МАГІСТЕРСЬКА РОБОТА

МР. ІПм - 14.00.00.000 ПЗ

Група ІПм-24-1

Гродецький Руслан

2025

Івано-Франківський національний технічний університет нафти і газу

Факультет інформаційних технологій

Кафедра інженерії програмного забезпечення

Гродецький Руслан Юрійович

(прізвище, ім'я, по батькові)

УДК 004.9

(індекс)

МАГІСТЕРСЬКА РОБОТА

Онтологічні моделі аналізу семантики природно мовного тексту

(назва роботи)

Інженерія програмного забезпечення

(назва освітньої програми)

121 - Інженерія програмного забезпечення

(шифр і назва спеціальності)

Гродецький Р.Ю.

(підпис, ініціали та прізвище здобувача освітнього ступеня)

Науковий керівник Юрчишин Володимир Миколайович, д.т.н., професор

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Допущено до захисту

Завідувач кафедри

доц. Бандура В.В.

(посада) (підпис) (дата) (ініціали та прізвище)

Нормоконтроль

доц. Вовк Р.Б.

(посада) (підпис) (дата) (ініціали та прізвище)

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

Івано-Франківськ – 2025

Івано-Франківський національний технічний університет нафти і газу

Факультет інформаційних технологій

Кафедра інженерії програмного забезпечення

Освітній рівень магістр

Спеціальність 121 – Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ:
Зав. кафедрою **ІПЗ**

доц. В.В. Бандура

“ 04 ” вересня 2025 р.

ЗАВДАННЯ

НА МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТУ

Гродецькому Руслану Юрійовичу

(прізвище, ім’я, по-батькові)

1. Тема магістерської роботи “ Онтологічні моделі аналізу семантики природно мовного тексту”

керівник проекту (роботи) Юрчишин В.М., д.т.н., професор

затверджені наказом закладу вищої освіти від “ 05 ” листопада 2025 р. № 695/7

2. Строк подання студентом проекту (роботи) 15 грудня 2025 р.

3. Вихідні дані до проекту (роботи) Теоретичні концепції та формальні моделі побудови та функціонування інформаційних технологій семантики природної мови користувача

4. Зміст розрахунково - пояснювальної записки(перелік питань, які потрібно розробити)

1. Дослідження предметної області застосування онтологій для аналізу тексту
2. Онтологічні моделі семантичного узгодження природно мовного тексту
3. Імплементація онтологічних моделей аналізу та узгодження семантики тексту
4. Розробка методики вимірювання гібридної рядкової подібності тексту

5. Перелік графічного матеріалу (з точним зазначенням обов’язкових креслень)

1. Приклад реалізації бази даних WordNet (рис. 1.1)
2. Мережева архітектура WordNet (рис. 1.2)
3. Онтологія веб каталогу Google (рис. 1.3)
4. Приклад онтології "Бізнес" (рис. 1.4)
5. Послідовність етапів процесу узгодження онтологій (рис. 1.5)

6. Консультанти розділів проекту (роботи)

Розділ	Консультант	Підпис, дата
Перевірка на плагіат	доц., к.т.н. Вовк Р.Б.	

7. Дата видачі завдання 04 вересня 2025 р.

Керівник _____
(підпис)

Завдання прийняв до виконання_____
(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назви етапів магістерської роботи	Строк виконання етапів роботи	Примітка
1	Підбір і вивчення літератури по темі магістерської роботи	15.09.2025	виконано
2	Дослідження предметної області застосування онтологій для аналізу тексту	29.09.2025	виконано
3	Онтологічні моделі семантичного узгодження природно мовного тексту	15.10.2025	виконано
4	Імплементація онтологічних моделей аналізу та узгодження семантики тексту	08.11.2025	виконано
5	Розробка методики вимірювання гібридної рядкової подібності тексту	20.11.2025	виконано
6	Реалізація функціональності запропонованої технології	01.12.2025	виконано
7	Затвердження пояснювальної записки роботи завідувачем кафедри	15.12.2025	виконано

Студент – магістр _____
(підпис)

Керівник роботи _____
(підпис)

АНОТАЦІЯ

Магістерська робота: 76 с., 15 рис., 5 табл., 38 джерел.

Тема: Онтологічні моделі аналізу семантики природно мовного тексту

Мета роботи - дослідження онтологічних моделей і методів аналізу семантики природномовного тексту, що забезпечують підвищення точності узгодження понять і формалізацію процесу семантичного.

Об'єкт дослідження - процеси семантичного аналізу та узгодження природномовних текстів у контексті онтологічного моделювання знань.

Предмет дослідження - онтологічні моделі, методи та метрики подібності, що використовуються для формалізації семантичних відношень і вимірювання подібності між текстовими елементами природної мови.

Результати дослідження

В роботі розроблено методику гібридної рядкової подібності, яка поєднує лексичні та семантичні показники. Запропонована методика була реалізована у вигляді програмного модуля, що забезпечує автоматичну оцінку подібності між текстовими фрагментами на рівні понять і термінів.

Висновок

Запропоновано методику гібридного вимірювання подібності, що інтегрує семантичні та рядкові показники. Удосконалено підхід до використання лексичної бази знань WordNet для побудови семантичних відношень між термінами природної мови.

ОНТОЛОГІЯ, СЕМАНТИЧНЕ УЗГОДЖЕННЯ, АНАЛІЗ ПРИРОДНОЇ МОВИ, WORDNET, СЕМАНТИЧНА ПОДІБНІСТЬ, ГІБРИДНА МЕТРИКА, ОБРОБКА ТЕКСТУ, КОНЦЕПТУАЛІЗАЦІЯ ЗНАНЬ.

ABSTRACT

Master Thesis: 76 pp., 15 fig., 5 tab., 38 sources.

Topic: Ontological models of natural language text semantic analysis

The aim of the work is to study ontological models and methods of natural language text semantic analysis, which ensure an increase in the accuracy of concept matching and formalization of the semantic process.

The object of the research is the processes of semantic analysis and matching of natural language texts in the context of ontological knowledge modeling.

The subject of the research is ontological models, methods and similarity metrics used to formalize semantic relations and measure similarity between text elements of natural language.

Research results

The paper developed a hybrid string similarity method that combines lexical and semantic indicators. The proposed method was implemented in the form of a software module that provides automatic assessment of similarity between text fragments at the level of concepts and terms.

Conclusion

A hybrid similarity measurement method that integrates semantic and string indicators is proposed. The approach to using the WordNet lexical knowledge base for constructing semantic relationships between natural language terms has been improved.

ONTOLOGY, SEMANTIC MATCHING, NATURAL LANGUAGE ANALYSIS, WORDNET, SEMANTIC SIMILARITY, HYBRID METRICS, TEXT PROCESSING, KNOWLEDGE CONCEPTUALIZATION.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

NLP - Natural Language Processing - Обробка природної мови

COMA - Composite Schema Matching System

ЛБЗ - Лексична база знань

OAEI - Ontology Alignment Evaluation Initiative

ОПМ - обробки природної мови

УОЗ - узгоджувачі на основі знань

Sim Similarity - подібність - загальна метрика

Comm – Commonality – спільність

Diff Difference - різниця

SimWN - WordNet Similarity - подібність на основі WordNet

LCS - Longest Common Substring - найбільший спільний підрядок

ВСТУП

Актуальність теми.

У сучасну епоху цифрової трансформації стрімке зростання обсягів текстової інформації створює потребу в ефективних засобах її автоматизованого аналізу та інтерпретації. Обробка природної мови (Natural Language Processing, NLP) стала однією з ключових галузей штучного інтелекту, що забезпечує взаємодію людини з комп'ютером на семантичному рівні. Однак, попри значні досягнення у створенні статистичних та нейромережевих моделей, проблеми формалізації знань і розуміння змісту текстів залишаються актуальними.

Онтологічні моделі виступають ефективним засобом подолання цього розриву між формальними структурами та семантичним контекстом природної мови. Вони дозволяють представити знання у вигляді концептуальних структур, що відображають об'єкти, відношення та властивості предметної області. Такий підхід уможливлює узгодження та інтеграцію різнорідних інформаційних ресурсів, а також підвищує точність семантичного аналізу текстів.

У роботі розглядаються теоретичні та прикладні аспекти побудови онтологічних моделей, формалізації процесу узгодження семантики, а також методики вимірювання семантичної подібності на основі лексичних баз знань, зокрема WordNet. Розроблені методи дозволяють удосконалити процеси розпізнавання, класифікації та порівняння текстових фрагментів у задачах інформаційного пошуку, машинного перекладу та інтелектуальних систем аналізу даних.

Актуальність дослідження зумовлена зростанням обсягів неструктурованих текстових даних у науковій, освітній, медійній та бізнесовій сферах, що потребують глибокого семантичного аналізу. Традиційні методи обробки текстів, засновані на статистичних підходах або

поверхневому порівнянні лексем, часто не забезпечують адекватного відображення смислових відношень між поняттями.

Онтологічне моделювання, навпаки, дозволяє формалізувати знання у вигляді системи понять і зв'язків, що відповідають природному семантичному простору мови. Це забезпечує можливість ефективного узгодження текстових структур, виявлення еквівалентних понять і подолання неоднозначностей у тлумаченні.

Особливої актуальності набуває проблема семантичного узгодження онтологій, що виникає в умовах інтеграції різних інформаційних систем та джерел знань. Розробка методів вимірювання семантичної подібності, які поєднують лексичні й контекстуальні ознаки, є необхідною умовою створення інтелектуальних систем нового покоління. Таким чином, дослідження є своєчасним і має важливе значення для розвитку технологій штучного інтелекту, машинного навчання та обробки природної мови.

Метою роботи є дослідження онтологічних моделей і методів аналізу семантики природномовного тексту, що забезпечують підвищення точності узгодження понять і формалізацію процесу семантичного.

Об'єктом дослідження є процеси семантичного аналізу та узгодження природномовних текстів у контексті онтологічного моделювання знань.

Предметом дослідження є онтологічні моделі, методи та метрики подібності, що використовуються для формалізації семантичних відношень і вимірювання подібності між текстовими елементами природної мови.

Для досягнення поставленої мети в роботі вирішувалися такі основні завдання:

- Провести аналіз предметної області застосування онтологій для семантичного аналізу тексту.

- Дослідити існуючі підходи до узгодження онтологій і визначити їх переваги та обмеження.

- Розробити формальну модель процесу узгодження та визначення кандидатів на відповідність між онтологічними елементами.

- Проаналізувати можливості застосування лексичних баз знань, зокрема WordNet, у задачах вимірювання семантичної подібності.

- Провести експериментальну перевірку запропонованих методів і оцінити їх ефективність для задач аналізу природномовного тексту.

У **роботі використано комплекс теоретичних і прикладних методів**, зокрема:

- методи формалізації знань і онтологічного моделювання — для опису предметної області та побудови концептуальних структур;

- методи лексико-семантичного аналізу — для визначення смислових зв'язків між поняттями природної мови;

- алгоритми вимірювання семантичної подібності на основі WordNet — для оцінювання зв'язків між поняттями у контексті;

- методи експериментального моделювання та порівняльного аналізу — для перевірки ефективності розроблених підходів.

Наукова новизна отриманих результатів полягає в удосконаленню методологічного підходу до аналізу природномовного тексту на основі онтологічного моделювання та розробці формальної моделі процесу семантичного узгодження між онтологічними елементами.

Практичне застосування результатів

Результати роботи можуть бути використані при розробці інтелектуальних систем обробки природної мови, систем семантичного пошуку, класифікації текстів та машинного перекладу. Розроблені моделі й методики можуть бути інтегровані у прикладні програмні платформи для автоматизованого аналізу великих обсягів текстових даних.

Структура магістерської роботи. Робота складається зі вступу, трьох розділів та висновків. Загальний обсяг роботи становить 76 сторінок, і містить 15 рисунків, 5 таблиць, список використаних джерел із 38 найменувань.

РОЗДІЛ 1. ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ ЗАСТОСУВАННЯ ОНТОЛОГІЙ ДЛЯ АНАЛІЗУ ТЕКСТУ

1.1. Роль та методи узгодження онтологій

Онтології являють собою фундаментальний компонент сучасних систем, орієнтованих на знання. Вони слугують уніфікованим джерелом спільно узгодженої та строго формалізованої термінології, що критично важливо для забезпечення інтероперабельності систем через ефективний обмін та повторне використання знань.

1.1.1. Проблема різноманітності та необхідність узгодження

Проте, множинність підходів до концептуалізації конкретної предметної області (домену) неминуче призводить до створення різних онтологій. Ці онтології часто містять суперечливі або частково (повністю) дублюючі (перекриваються) фрагменти. Ця несумісність створює значні перешкоди для безшовної інтеграції даних.

З огляду на це, узгодження онтологій (ontology alignment або ontology matching) є необхідною та ключовою технікою. Його основна мета — встановлення відповідностей (мапінгу) між сутностями (класами, властивостями, індивідами) різних онтологій. Успішне узгодження забезпечує семантичну сумісність, що є передумовою для ефективної інтеграції та спільного використання гетерогенних даних.

1.1.2. Методологічні підходи до узгодження онтологій

Процес узгодження онтологій зазвичай спирається на комплекс евристичних методів, які оцінюють ступінь подібності між сутностями. Серед них виділяють два важливі класи методів, які застосовуються для порівняння лексичних елементів (термінів):

1. Методи семантичної подібності

Ці методи ґрунтуються на аналізі значеннєвої (семантичної) близькості між термінами. Для кількісного визначення семантичної відстані часто використовуються лексичні бази знань, такі як WordNet.

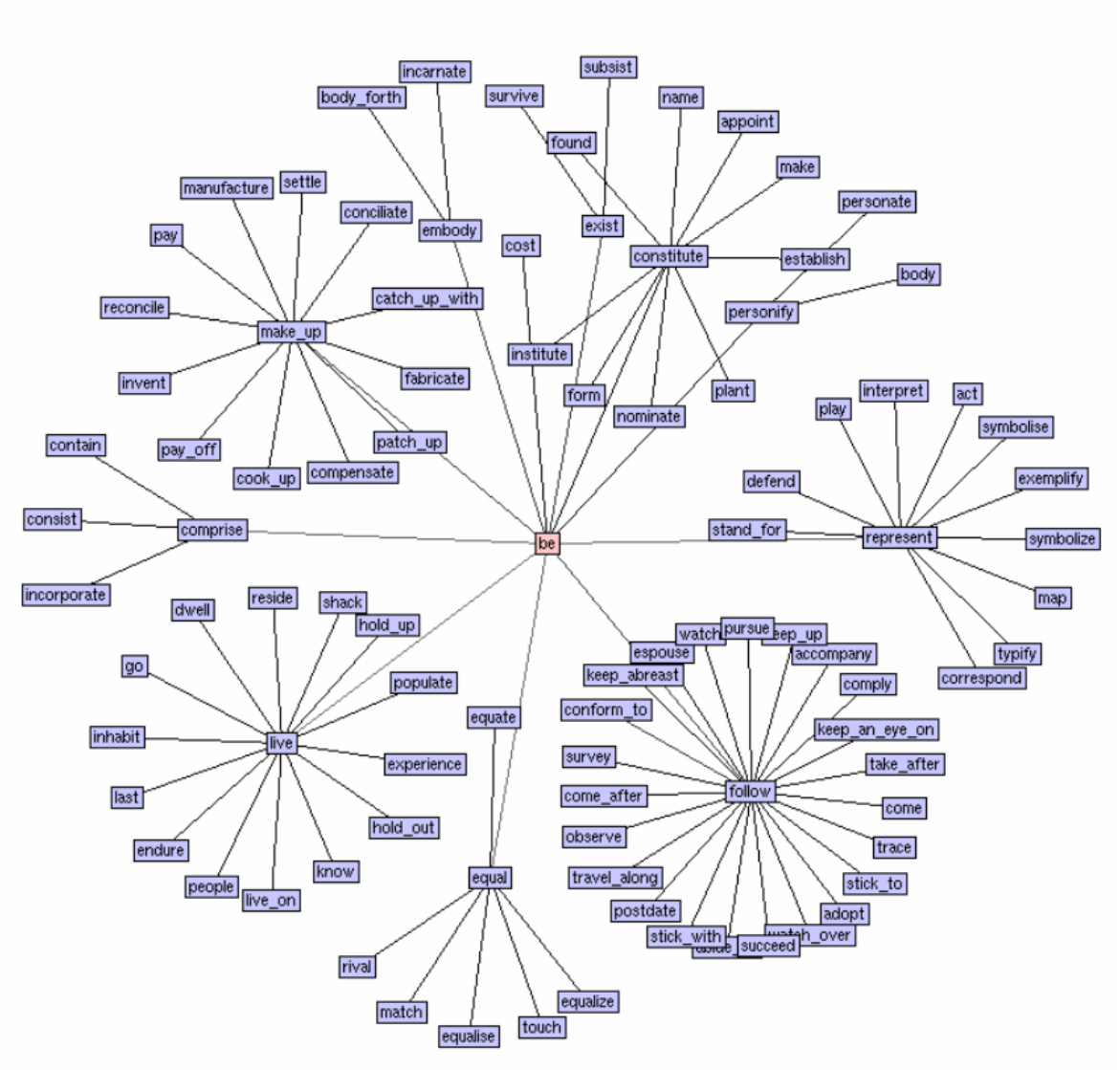

Рис. 1.1. Приклад реалізації бази даних WordNet

WordNet, будучи великою лексичною базою даних англійської мови (хоча існують аналоги для інших мов), організовує слова у синонімічні набори (synsets), пов'язані концептуально-семантичними та лексичними зв'язками. Це дозволяє обчислити семантичну подібність між двома термінами, виходячи з їхнього розташування та зв'язків у цій ієрархічній структурі.

2. Методи на основі рядкових метрик

Ці методи оцінюють подібність між двома текстовими рядками (назвами сутностей) на основі їхньої синтаксичної (побудовної) близькості. Вони використовують метрики рядкової відстані (англ. string distance metrics) для вимірювання мінімальної кількості операцій (вставки, видалення, заміни символів), необхідних для перетворення одного рядка в інший, або для визначення кількості спільних/відмінних характеристик.

Зокрема, відстань Джаро-Вінклера (Jaro-Winkler distance) є однією з високоточних метрик у цьому класі. Вона модифікує метрику Джаро, надаючи перевагу відповідностям на початку рядка (префіксу), що особливо корисно для порівняння назв, де відмінності часто виникають у кінці. Ця метрика обчислює подібність, виходячи з кількості спільних символів та транспозицій (перестановок) між порівнюваними словами, ефективно відображаючи спільні та відмінні структурні риси лексичних елементів.

У даній роботі для визначення семантичної подібності було застосовано лексичну базу даних WordNet, тоді як для визначення подібності на основі рядкових метрик було використано відстань Джаро-Вінклера.

Інтеграція та комбінування результатів, отриманих за допомогою семантичних та рядкових метрик, у рамках гібридного підходу дозволяє значно підвищити точність та повноту процесу узгодження онтологій, забезпечуючи надійну основу для об'єднання знань у розподілених інформаційних системах.

1.2. Концептуальні основи онтологій та проблема узгодження

Онтологія є фундаментальним інструментом для формалізації знань, що забезпечує словник для опису певної предметної області (домену) та формальну специфікацію значень термінів, які входять до цього словника. Залежно від рівня строгості та точності цієї специфікації, поняття онтології охоплює широкий спектр концептуальних моделей і структур даних, від

простих класифікацій та схем баз даних до повністю аксіоматизованих теорій.

Узгодження онтологій (Ontology Alignment або Matching) є ключовим механізмом забезпечення інтероперабельності (взаємодії) в архітектурі Семантичного Вебу [1]. Крім того, ця техніка має важливе прикладне значення у традиційних завданнях інтеграції даних, спрямованих на вирішення проблеми семантичної гетерогенності.

Для побудови спільного Семантичного Вебу, який би дозволяв обмін та повторне використання даних між різними застосунками, підприємствами та спільнотами, критично необхідно розробити ефективні методи для порівняння, узгодження та інтеграції різних онтологій. Сучасні системи узгодження онтологій використовують різноманітні стратегії для визначення ступеня подібності між сутностями, включаючи, але не обмежуючись: подібність рядків (лексичні методи), використання синонімів (семантичні методи), структурну подібність та порівняння на основі екземплярів (даних).

1.2.1. Роль лексичних баз знань

Лексична база знань (ЛБЗ) у сфері інформатики, лінгвістики та обробки природної мови (ОПМ) – це спеціалізований тип бази знань, який містить формально структуровану інформацію про лексичний склад мови та семантичні відношення між словами.

По суті, ЛБЗ є спробою змоделювати та представити в машиночитному форматі знання, що традиційно містяться у словниках, тезаурусах та інших лінгвістичних довідниках, але з додаванням явних, структурованих зв'язків.

Ключові характеристики ЛБЗ:

- Формалізація лексики. Містить перелік слів (лексичних одиниць) мови або певної предметної області.

- Специфікація значення (семантика). Надає точні визначення лексичних значень слів, часто вирішуючи проблему багатозначності

(полісемії), коли одне слово має кілька значень (наприклад, "крило" літака та "крило" птаха).

- Структуровані відношення. Головна відмінність від звичайного словника – фіксація відношень між словами чи поняттями.

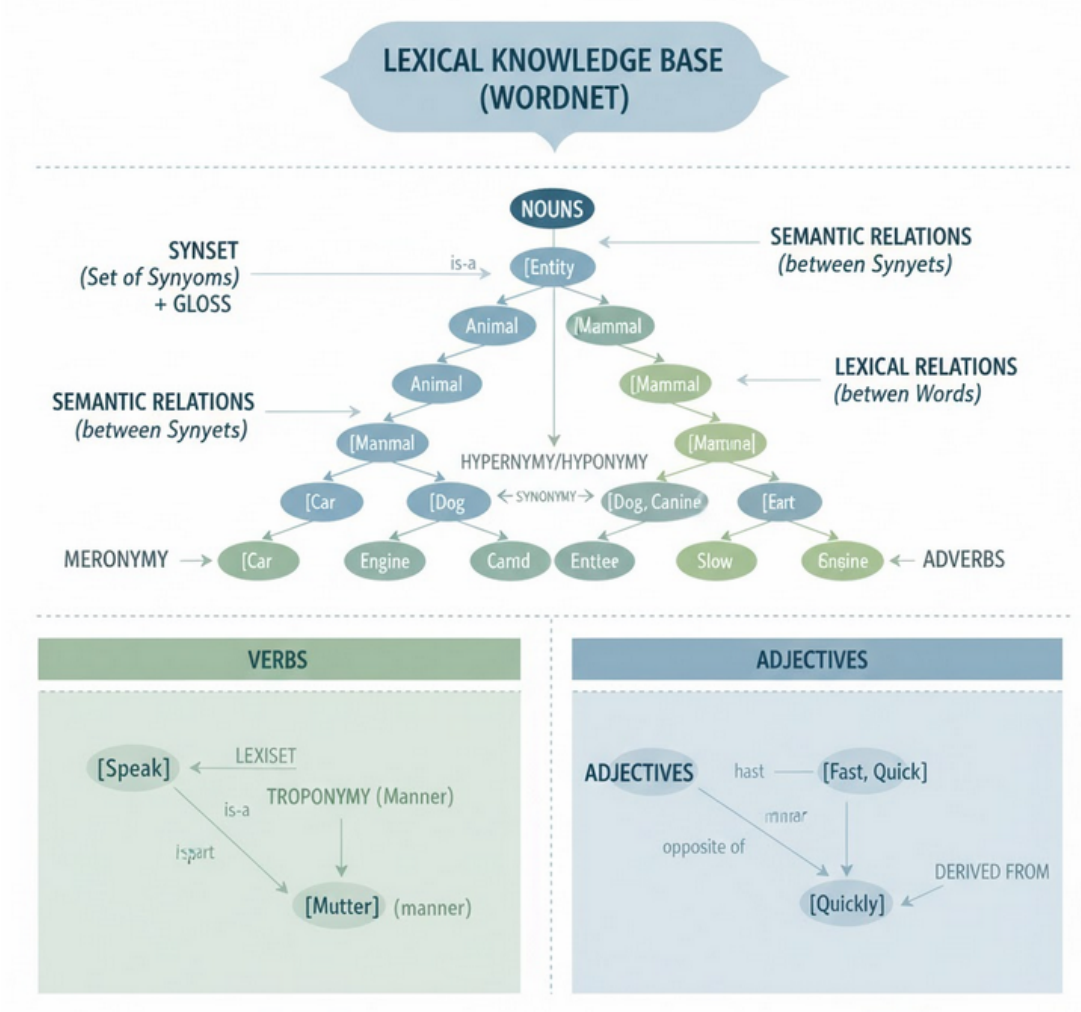

Рис. 1.2. Мережева архітектура WordNet

WordNet функціонує як орієнтований ациклічний граф (Directed Acyclic Graph, DAG), де:

- вузли (Nodes) представляють синсети.

ребра (Edges) представляють семантичні та лексичні відношення.

Завдяки цій структурі, WordNet дозволяє програмно обчислювати семантичну подібність між двома словами, вимірюючи, наскільки вони

близькі в ієрархії графа (наприклад, за довжиною шляху до їхнього найближчого спільного гіпероніма).

Використання синонімів є важливим для вирішення проблеми, коли різні терміни використовуються в онтологіях для позначення одного і того ж поняття. WordNet [2] є авторитетною лексичною базою даних для англійської мови, яка групує слова в набори синонімів (synsets), надає їхні короткі визначення та приклади вживання, а також фіксує семантичні та лексичні відношення між цими наборами або їхніми членами. Таким чином, WordNet функціонує як комбінація словника та тезаурусу. Завдяки своїй структурі, WordNet може значно підвищити ефективність метрик подібності та широко використовується для кількісного вимірювання семантичної близькості.

1.2.2. Огляд методології дослідження

Представлене дослідження зосереджується на проблематиці узгодження онтологій, приділяючи особливу увагу лексичному (рядковому) та семантичному узгодженню двох лексичних одиниць (слів).

1. Семантичне узгодження

Для встановлення семантичної подібності між двома словами використано підхід, заснований на метриках подібності WordNet. Цей підхід реалізує функцію порівняння, яка, використовуючи внутрішню структуру WordNet, обчислює ступінь семантичного зв'язку між заданими словами. Результат являє собою числове значення в діапазоні від 0 до 1, де вище значення відповідає більшій семантичній близькості.

2. Рядкове (лексичне) узгодження

Узгодження рядків є альтернативним засобом вимірювання подібності, який ґрунтується на аналізі спільних та відмінних підрядків у порівнюваних словах. З метою підвищення точності та надійності результатів, отриманих на основі елементарного аналізу підрядків, до загального показника подібності була інтегрована відстань Джаро-Вінклера (Jaro-Winkler distance). Ця метрика ефективно враховує як спільні, так і відмінні структурні риси

лексичних одиниць, сприяючи отриманню більш адекватного кількісного відображення подібності.

У рамках даної роботи проведено аналіз та порівняння результатів, отриманих за допомогою обох технік (семантичної та рядкової), для виявлення сильних та слабких сторін кожної методики у визначенні подібності та відмінності між двома лексичними одиницями.

1.3. Архітектура онтологій та формалізація концептуалізації домену

Онтологія у цьому контексті визначається як формальний спосіб специфікації концептуалізації конкретної прикладної області (домену). Ця специфікація здійснюється через визначення набору концептів (класів), атрибутів та відношень, виражених у формальній мові.

1.3.1. Структурні компоненти онтології

Онтологічна модель будується на трьох ключових елементах:

1. Концепти (Concepts/Classes) - абстрактні категорії або сутності, які існують у домені.

2. Атрибути (Attributes) - властивості, що описують характеристики концептів.

3. Відношення (Relations) - зв'язки між концептами, які можуть бути визначені користувачем або належати до набору заздалегідь визначених (предифінованих) відношень із відомою семантикою.

Фундаментальні відношення, що використовуються для структуризації онтологій, включають:

- Is-A (Підклас-Надклас) - відношення гіперонімії/гіпонімії, що вказує на ієрархію успадкування.

- Part-Of (Частина-Ціле) - відношення меронімії/холонімії, що вказує на структурну композицію.

- Instance-Of (Екземпляр-Клас) - відношення, що пов'язує конкретний об'єкт даних (індивіда) з його концептом (класом).

1.3.2. Концептуальна ієрархія як спрощена онтологія

Концептуальна ієрархія є спрощеною формою онтології, яка обмежується лише набором концептів і виключно відношеннями Is-A між ними, не включаючи атрибути.

Наприклад, можна розглядати ієрархію веб-каталогу (як показано на рис. 1.3) як базову концептуальну ієрархію, що складається лише з класів, пов'язаних відношеннями Is-A.

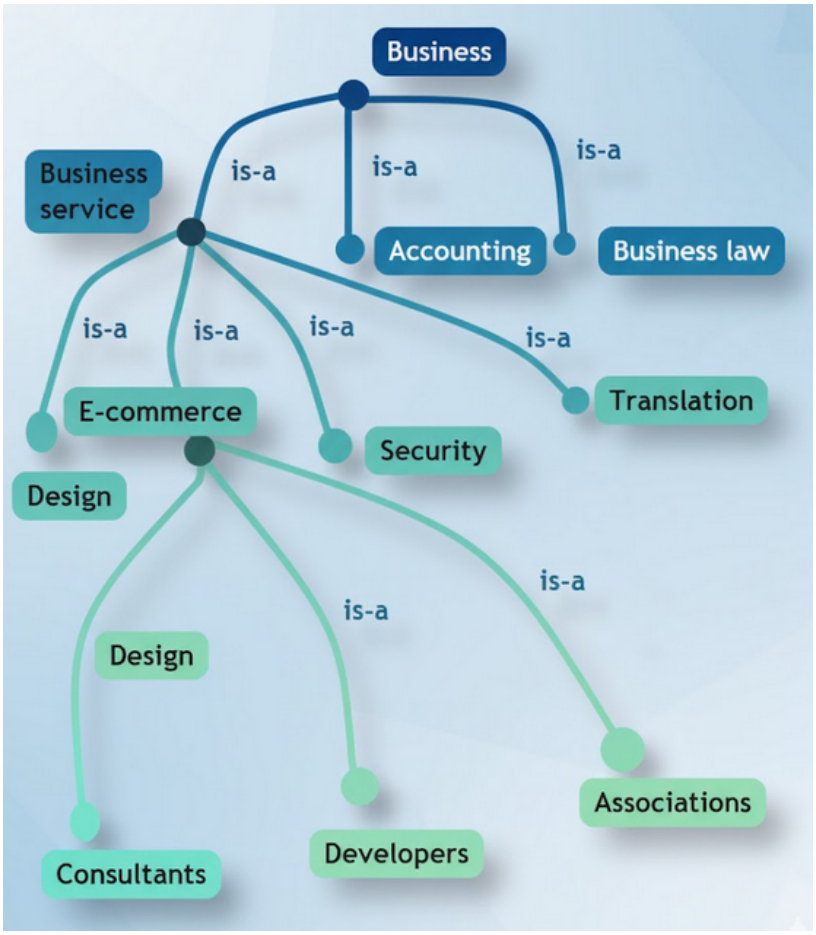

Рис. 1.3. Онтологія веб каталогу Google

Для побудови повноцінної онтології з базової концептуальної ієрархії необхідно додати атрибути та екземпляри даних.

Ілюстративний приклад (рис. 1.4) демонструє онтологію "Бізнес", яка розширює концептуальну ієрархію, додаючи атрибути до концепту Association (Асоціація):

- Концепт: Association (Асоціація).

- Атрибути: BN (Бізнес-Номер), City (Місто), Street (Вулиця), Zip (Поштовий індекс).

- Екземпляри (Data Instances): B8 та B2, які є конкретними реалізаціями концепту Association.

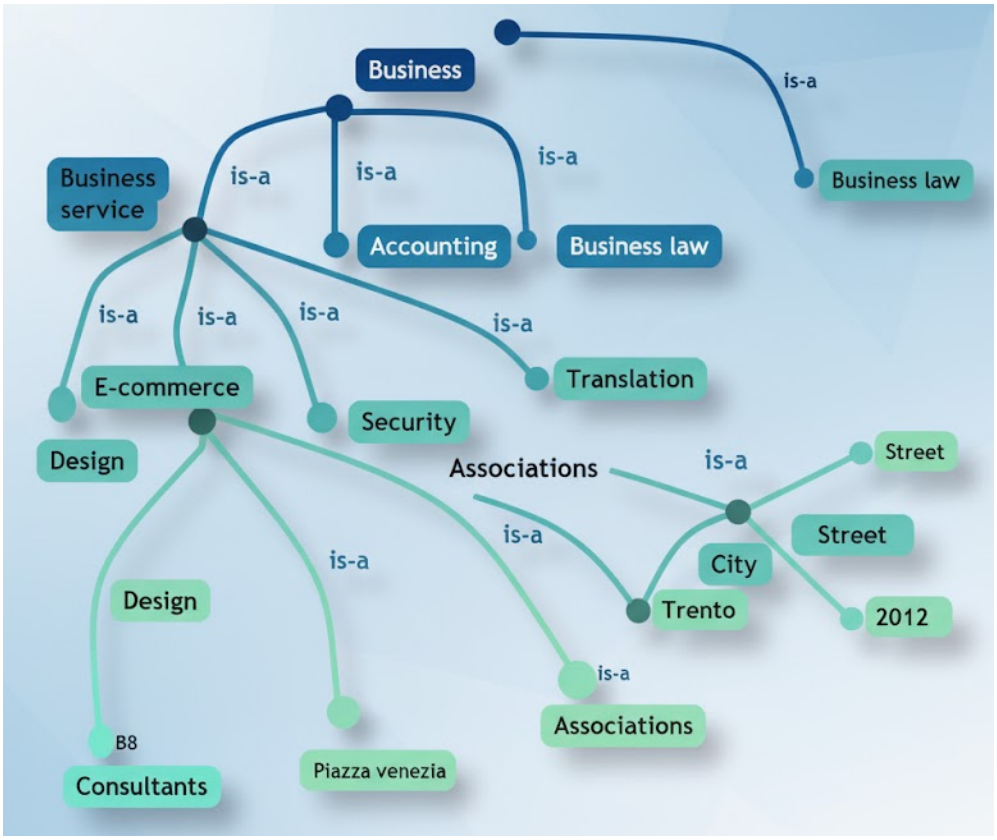

Рис. 1.4. Приклад онтології "Бізнес"

Кожен екземпляр має визначений набір значень для своїх атрибутів, наприклад екземпляр B8 має значення: BN = "B8", City = "Trento", Street = "Piazza Venezia", і т.д.

Така структурована модель дозволяє не лише класифікувати сутності (через Is-A), але й детально описувати їхні властивості (через атрибути) та фіксувати конкретні дані (через екземпляри).

1.4. Теоретичні засади та методологічні підходи узгодження онтологій

Узгодження (узгоджування) сутностей (Matching) є критично важливою операцією в низці предметних областей, що охоплюють Семантичний Веб, інтеграцію схем та онтологій, сховища даних, електронну комерцію та посередництво запитів.

1.4.1. Формалізація процесу узгодження

Процес узгодження приймає на вхід дві схеми або онтології, кожна з яких складається з дискретного набору сутностей (наприклад, таблиць, XML-елементів, класів, властивостей, правил, предикатів). Його результатом є ідентифікація відношень (таких як еквівалентність або підпорядкованість), що існують між цими сутностями.

Проблема узгодження онтологій є, по суті, завданням зіставлення ієрархій концептів і відношень двох різних онтологічних моделей. Якщо онтологія не містить ієрархії відношень, вона може бути розглянута як схема. У сфері узгодження схем виконано значний обсяг досліджень, особливо в контексті інтеграції та перетворення даних [3].

Незважаючи на широку поширеність та критичне значення, узгодження онтологій досі значною мірою виконується вручну. Цей процес є трудомістким, ресурсоємним та схильним до помилок, що створює основне вузьке місце у створенні великомасштабних систем управління інформацією.

Стрімкий розвиток таких технологій, як Всесвітня павутина (WWW), XML та Семантичний Веб, лише посилює попит на ефективний обмін інформацією і, відповідно, загострює проблему масштабування узгодження. Отже, розробка автоматизованих інструментів для підтримки процесу узгодження онтологій набула критичного значення для успішної реалізації широкого спектра інформаційних систем.

1.4.2. Огляд існуючих підходів до узгодження онтологій

Хоча сфера узгодження онтологій є відносно новою, існує низка значущих досліджень

Еволюція системи LSD, що використовує методи машинного навчання для напівавтоматичного знаходження відображень схем для інтеграції даних. Вона передбачає фазу навчання на екземплярах даних онтологій, виявлення характерних шаблонів та правил відповідності. Прогнози, отримані від окремих алгоритмів, комбінуються мета-навчальником для формування остаточних результатів.

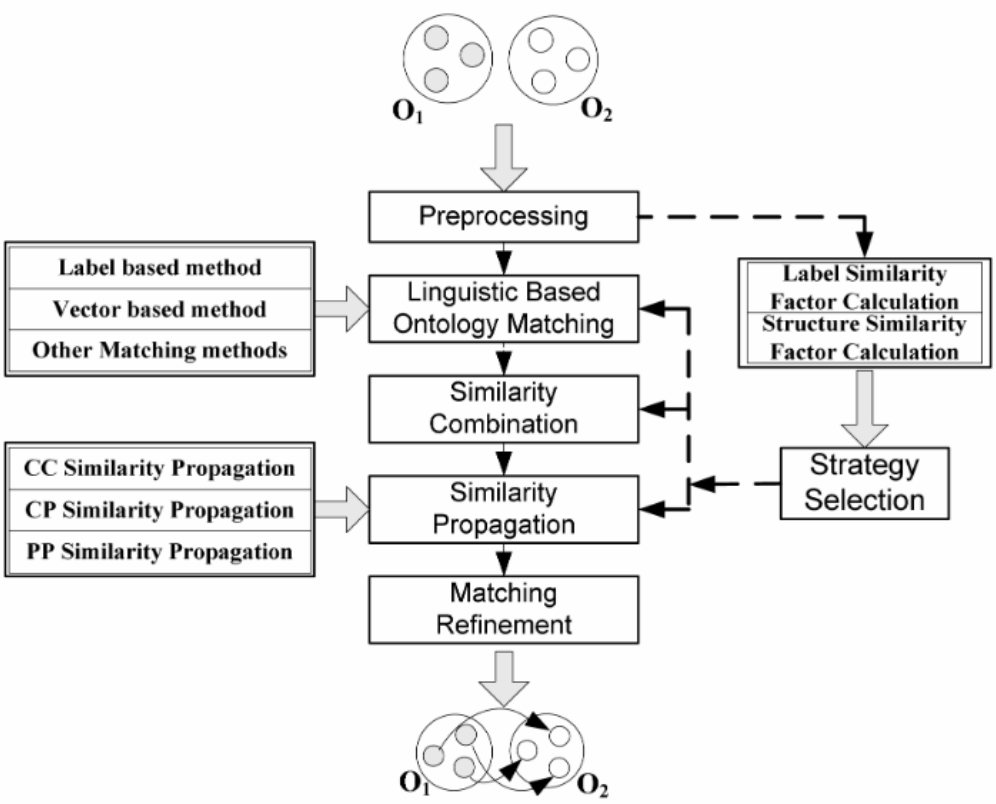

Рис. 1.5. Послідовність етапів процесу узгодження онтологій

На рис. 1.5 представлено послідовність етапів процесу узгодження онтологій, реалізованого в системі RiMOM (Robust Ontology Matching):

1. Попередня обробка (Preprocessing)

На цьому етапі для двох вхідних онтологій виконується генерація дескрипторів для кожної сутності. Далі відбувається обчислення двох

факторів подібності, які будуть використані для динамічного керування процесом на подальших кроках.

2. Узгодження онтологій на основі лінгвістичних методів (Linguistic-based Alignment)

Цей крок передбачає виконання множинних стратегій, що ґрунтуються на лінгвістичному аналізі. Кожна стратегія використовує різні типи онтологічної інформації та генерує окремий показник подібності для кожної пари сутностей. Ці стратегії динамічно обираються для включення у відповідні завдання узгодження.

3. Комбінування показників подібності (Similarity Combination)

На цьому етапі відбувається інтеграція (комбінування) результатів подібності, отриманих від обраних лінгвістичних стратегій. Вагові коефіцієнти у формулі комбінування визначаються на основі двох факторів подібності, обчислених на етапі попередньої обробки.

4. Поширення подібності (Similarity Propagation)

Цей етап враховує структурну подібність між онтологіями. Для цього використовуються три стратегії поширення подібності:

- Концепт-до-Концепту (Concept-to-Concept)
- Властивість-до-Властивості (Property-to-Property)
- Концепт-до-Властивості (Concept-to-Property)

5. Генерація та уточнення узгодження (Alignment Generation and Refinement)

На завершальному етапі виконується фінальне налаштування та вивід результату узгодження (фінального мапінгу).

Як видно на рис. 1.5, механізм вибору стратегій (strategy selection) застосовується на трьох з п'яти етапів (крок 2, крок 3 та крок 4). Він визначає:

1. Яка інформація має бути використана у лінгвістично-орієнтованій стратегії (Крок 2).

2. Які вагові коефіцієнти слід застосовувати при комбінуванні показників подібності (Крок 3).

3. Яку саме стратегію поширення подібності необхідно використовувати (Крок 4).

Anchor-PROMPT [5] - це алгоритм який будує спрямований розмічений граф, що моделює онтологію через ієрархії концептів (класів) та відношень (слотів). Вхідними даними є початковий набір якорів (пар пов'язаних концептів), визначених користувачем або лексичним узгодженням. Anchor-PROMPT аналізує шляхи в підграфі, обмеженому якорями, і на основі частоти появи концептів у подібних позиціях на подібних шляхах визначає їхню семантичну подібність.

Anchor-PROMPT є інструментом для графового відображення та узгодження онтологій, заснованим на алгоритмі, що експлуатує структуру графа онтологій.

Алгоритм Anchor-PROMPT шукає кореляції між концептами двох онтологій шляхом паралельного обходу шляхів заданої довжини. Довжина шляху (кількість ребер) задається користувачем і обмежується початковими та кінцевими точками (originating and terminating points) у відповідних підграфах. Початкові точки називаються якорями (anchors).

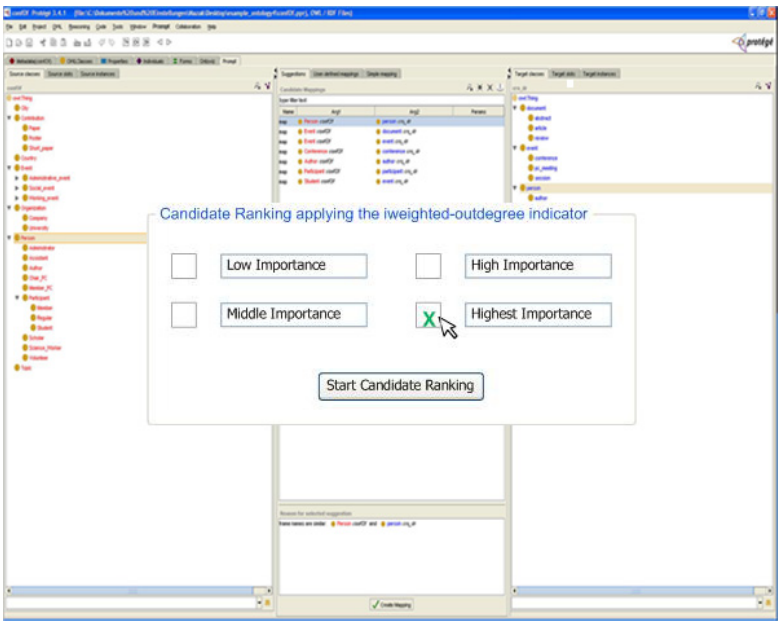

Рис. 1.6. Приклад запиту для ранжування концептів за допомогою Anchor-PROMPT

На рис. 1.6 показано приклад запиту для ранжування концептів за допомогою Anchor-PROMPT. Це може бути реалізовано шляхом конфігурації віджету PROMPT Tab у середовищі Protégé.

Такий підхід дозволяє легко та швидко знаходити відповідні початкові якірні пари через просту взаємодію "вкажи і клацни" (point-and-click). Як наслідок, зменшується навантаження на користувача, якому більше не потрібно самостійно аналізувати складну структуру та концепти вихідних онтологій для визначення лише початкових і кінцевих точок шляху.

Таким чином, Anchor-PROMPT бере до уваги нелокальний контекст сутностей, аналізуючи їхнє розташування у загальній структурі онтології.

SAT [7] - метод, що приймає на вхід два графи концептів і встановлює між ними відношення (еквівалентність, перекриття, невідповідність, більш загальні/специфічні). Ключова ідея полягає у використанні логіки та SAT-вирішувача.

Концепція вузла кодується як кон'юнкція всіх концептів вузлів на шляху від кореня і трансформується у пропозиційну формулу. Відношення, яке необхідно довести, також перетворюється у пропозиційну формулу. SAT-вирішувач перевіряє істинність припущеного відношення на основі обчисленого набору формул.

Спочатку онтології розроблялися з метою вичерпного визначення всіх концептів у межах доменної області та їхніх відношень. Яскравим прикладом популярної лексичної онтології є WordNet [2], яка моделює лексичні знання носія англійської мови.

Дане дослідження спрямоване на застосування технік онтологічного узгодження, зокрема метрик подібності, для обчислення ступеня подібності між описами концептів або відношень двох слів. У роботі акцентовано увагу на семантичному порівнянні (з використанням лексичної бази даних WordNet) та лексичному (рядковому) порівнянні (наприклад, метрика Джаро-Вінклера).

1.5. WordNet як лексична база знань

Спочатку онтології використовувалися для вичерпного визначення концептів і відношень у домені. Прикладом популярної онтології, що моделює лексичні знання, є WordNet.

WordNet є зовнішнім термінологічним словником та лексичною базою даних англійської мови, що включає іменники, дієслова, прислівники та прикметники. Розроблений у Принстонському університеті під керівництвом Джорджа Армітейджа Міллера, WordNet організовує інформацію навколо лексичних груп, званих синсетами, та семантичних вказівників.

Синсет (Synset) неформально представляє набір слів-синонімів, що позначають одне конкретне поняття (наприклад, "benefit" та "profit" об'єднані в один синсет). Семантичний вказівник моделює відношення між цими синсетами. WordNet також надає короткі описи (глоси) та приклади вживання слів.

Таблиця 1.1.

Основні семантичні відношення WordNet

Відношення	**Опис**	**Приклад**
Гіперонім	Є узагальненням (відношення роду)	Моторний транспортний засіб є гіперонімом автомобіля
Гіпонім	Є видом (відношення виду)	Автомобіль є гіпонімом моторного транспортного засобу
Меронім	Є частиною (відношення частина-ціле)	Замок є меронімом дверей
Холонім	Містить частину (відношення ціле-частина)	Двері є холонімом замка
Тропонім	Є способом (відношення способу виконання, для дієслів)	Літати є тропонімом подорожі
Антонім	Протилежність	Залишатися на місці є антонімом подорожі

Атрибут	Атрибутивна характеристика	Швидкий є атрибутом швидкості
Зумовлення	Логічно зумовлює	Дзвінок по телефону зумовлює набирання номера
Причина	Викликає наслідок	Заподіяти біль викликає страждання

Заходи подібності, засновані на описі, часто називають термінологічними заходами подібності, оскільки вони використовують інформативні дескриптори. WordNet є ефективним інструментом для вимірювання подібності, оскільки усі синсети пов'язані між собою семантичними відношеннями, такими як гіпероніми/гіпоніми (відношення спеціалізації між категоріями).

Висновки до розділу

У першому розділі проведено всебічний аналіз предметної області застосування онтологій для аналізу природномовного тексту. З’ясовано, що онтології є фундаментальним інструментом формального представлення знань, який дозволяє систематизувати поняття та встановлювати семантичні зв’язки між ними. Визначено основні проблеми узгодження онтологій, зумовлені різноманітністю структур, форматів та концептуальних підходів. Розглянуто сучасні методології узгодження, зокрема логічні, структурні та лексико-семантичні, що забезпечують уніфікацію онтологічних моделей. Особливу увагу приділено ролі лексичних баз знань, зокрема WordNet, які виступають основою для побудови онтологій та формування семантичних зв’язків. Отже, у результаті дослідження визначено концептуальні та архітектурні засади, необхідні для побудови ефективних моделей семантичного узгодження тексту.

РОЗДІЛ 2. ОНТОЛОГІЧНІ МОДЕЛІ СЕМАНТИЧНОГО УЗГОДЖЕННЯ ПРИРОДНО МОВНОГО ТЕКСТУ

2.1. Теоретичні основи та методологія семантичного узгодження

Оператор узгодження (Matching Operator) виконує ключову функцію, приймаючи на вхід дві графоподібні структури (схеми або онтології) та встановлюючи відображення (mappings) між вузлами графів, які є семантично еквівалентними або пов'язаними.

2.1.1. Значимість узгодження в інформаційних системах

З появою Семантичного Вебу та постійним зростанням кількості гетерогенних джерел даних, переваги використання онтологій набувають все більшого визнання. Сфера застосування узгодження знань розширюється, охоплюючи широкий спектр завдань: від розпізнавання лексичного значення слів до пошуку біологічних макромолекул (таких як ДНК і білки).

У цьому дослідженні увага зосереджена на схемно-орієнтованому рішенні, тобто на системі узгодження, яка використовує виключно інформацію схеми (структури), не залучаючи екземпляри даних. Ми дотримуємося інноваційного підходу, відомого як семантичне узгодження [1].

Семантичне узгодження — це техніка, що використовується в комп'ютерних науках для ідентифікації інформаційних сутностей, пов'язаних семантично. Цей підхід базується на двох ключових концептуальних ідеях:

1. Обчислення відношень замість коефіцієнтів. Ми обчислюємо відображення між елементами схеми, встановлюючи семантичні відношення (наприклад, еквівалентність, більша загальність, розбіжність), на противагу традиційним підходам, які обчислюють коефіцієнти подібності у діапазоні [0,1].

2. Аналіз значення та формальна логіка. Семантичні відношення визначаються шляхом аналізу значення (концептів, а не міток), закодованого в елементах і структурах схем. Зокрема, мітки вузлів, написані природною мовою, переводяться у пропозиційні формули, які явно кодують призначене значення міток. Це дозволяє трансформувати проблему узгодження у проблему пропозиційної незадоволеності, яка ефективно вирішується за допомогою сучасних вирішувачів пропозиційної задоволеності (SAT).

2.1.2. Огляд пов'язаних досліджень

На сьогодні існує низка напівавтоматичних систем узгодження схем. Детальний огляд та класифікація підходів наведені в [15 – 18], а розширена класифікація, орієнтована на схеми та перспективи користувача, представлена у [13].

Розглянемо критерії класифікації алгоритмів узгодження. У [13] запропоновано критерії, що деталізують рівень узгодження елементів та структури:

1. Синтаксичні техніки.

Інтерпретують вхідні дані як функцію їхніх структур, використовуючи чітко сформульовані алгоритми (наприклад, ітеративне обчислення нерухомої точки для узгодження графів).

2. Зовнішні техніки.

Використовують зовнішні доменні або загальні ресурси знань (наприклад, WordNet).

3. Семантичні техніки.

Використовують формальну семантику (наприклад, семантику теорії моделей) для інтерпретації вхідних даних та обґрунтування результатів.

Класифікація систем за результатами узгодження розширює попередню класифікацію [3], розрізняючи системи за типом результату, що надається кінцевому користувачеві:

1. Відображення як рішення.

Системи розглядають узгодження як проблему оптимізації, а відображення є її оптимальним рішенням [9].

2. Відображення як теореми.

Системи покладаються на формальну семантику і вимагають, щоб відображення задовольняло її (наприклад, підхід, використаний у цій роботі).

3. Відображення як підказки подібності.

Системи генерують розумні підказки або оцінки подібності, які користувач остаточно верифікує [7].

2.1.3. Приклади сучасних схемно-орієнтованих систем узгодження

Узгодження елементів двох схем даних або екземплярів даних відіграє ключову роль у багатьох галузях, таких як сховища даних (data warehousing), електронний бізнес (e-business) та біохімічні застосунки.

Внутрішня модель даних, яку ми використовуємо для представлення моделей та відображень (mappings), базується на концепції спрямованих розмічених графів (directed labeled graphs).

Кожне ребро у графі представлено у вигляді трійки (s;p;o), де:

- s (Source): Вузол-джерело.

- o (Target): Вузол-призначення.

- p (Label): Мітка (або предикат) ребра, що описує відношення між вузлами s і o.

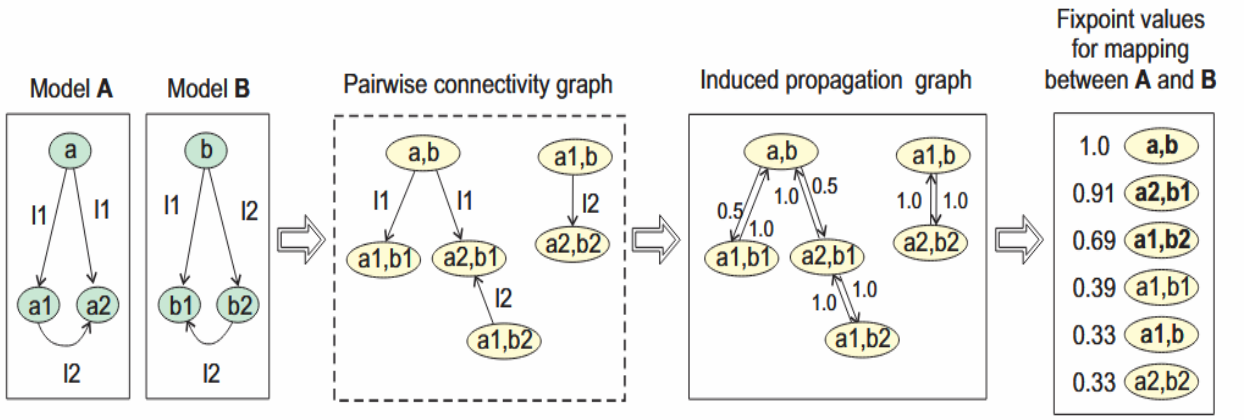

Рис. 2.1. Графічне представлення Similarity Flooding Algorith

Цей формат забезпечує формальне визначення моделі даних. Для ілюстрації роботи нашого алгоритму ми скористаємося простим прикладом, наведеним на рис. 2.1. Його верхня ліва частина містить дві моделі, A та B, які є вхідними даними для процесу узгодження.

RONDO (Similarity Flooding, SF):

- Алгоритм: Гібридний, заснований на поширенні подібності (Similarity Flooding).

- Техніки: Використовує лише синтаксичні техніки на рівні елементів та структури. Починається зі строкового порівняння міток вузлів, яке уточнюється через обчислення нерухомої точки.

- Результат: Розглядає відображення як рішення проблеми оптимізації.

CUPID - реалізує гібридний алгоритм узгодження, що включає синтаксичні техніки на рівні елементів.

Комбінує синтаксичні техніки на рівні елементів (строкове порівняння, префікси/суфікси) та структури (узгодження дерев, зважене за листками). Використовує зовнішні ресурси (попередньо скомпільований тезаурус).

Результат належить до категорії відображень як підказок подібності.

Навіть найбільш досконалі алгоритми узгодження схем генерують значну кількість помилок, особливо у випадку повністю автоматичних систем, де відсутня участь людини-проектувальника. Незважаючи на ці неточності, існують застосунки, які можуть використовувати результати узгодження без додаткового ручного коригування. Це можливо, коли достатньо досягнення найкращого можливого результату (best-effort matching) або коли відображення (matches) лише неявно впливають на кінцевий результат для користувача.

Розглянемо два сценарії автоматичного заповнення HTML-форм, які ілюструють це використання:

1. Автоматичне заповнення персональних даних у браузері

Більшість сучасних веб-браузерів пропонують функцію автоматичного заповнення форм (наприклад, введення імені та адреси перед здійсненням

покупки). Цей процес можна змоделювати як завдання узгодження, де схема базової веб-форми зіставляється з моделлю даних користувача, що зберігається в браузері.

Користувач очікує, що браузер зробить найкращу спробу (best-effort) заповнити особисті дані, які потім користувач підтверджує перед відправленням форми. Точність узгодження в цьому випадку не є абсолютно критичною, оскільки фінальна верифікація лежить на користувачеві.

2. Доступ до прихованого вебу за допомогою краулерів

Узгодження схем було запропоновано як засіб доступу до контенту, прихованого за HTML-формами (deep web). Краулер прихованого вебу (deep-web crawler) функціонує наступним чином:

1. Ідентифікація домену. Коли краулер зустрічає HTML-форму, він ідентифікує домен, до якого належить ця форма.

2. Узгодження зі схемою-посередником. Краулер узгоджує поля вводу форми з елементами попередньо обчисленої схеми-посередника (mediated schema) для цього домену (рис. 2.2).

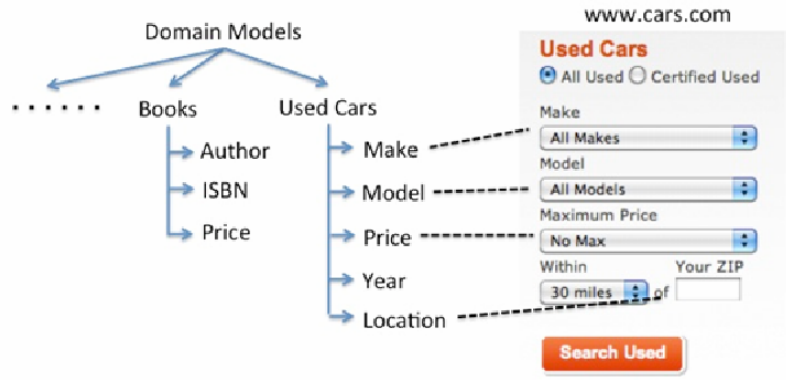

Рис. 2.2. Зіставлення між моделями предметної області та вхідними даними форми можна використовувати для автоматичного заповнення HTML-форм

3. Генерація запитів. Використовуючи зразки значень (на основі відомих значень елементів у схемі-посереднику), краулер генерує відправлення форм, конструюючи відповідні URL-адреси.

4. Індексація. Отримані сторінки додаються до індексу пошукової системи.

У цьому сценарії результати узгодження є проміжними результатами багатоетапного процесу. Кінцеві користувачі, ймовірно, не знають і не переймаються якістю безпосередньо результату узгодження, оскільки їх цікавить лише те, наскільки ефективно краулер використав базовий веб-сайт для розширення пошукового індексу.

Система узгодження схем COMA (COmbination of MAtching algorithms) це платформа для гнучкого комбінування множинних зіставляючих (matchers) алгоритмів. Система пропонує широкий спектр індивідуальних алгоритмів узгодження, включаючи інноваційний підхід, орієнтований на повторне використання результатів попередніх операцій узгодження.

COMA використовується як науковий фреймворк для всебічної оцінки ефективності різних алгоритмів узгодження та їхніх комбінацій на реальних схемах. Отримані результати на сьогоднішній день демонструють перевагу комбінованих підходів до узгодження і вказують на високу цінність стратегій, орієнтованих на повторне використання.

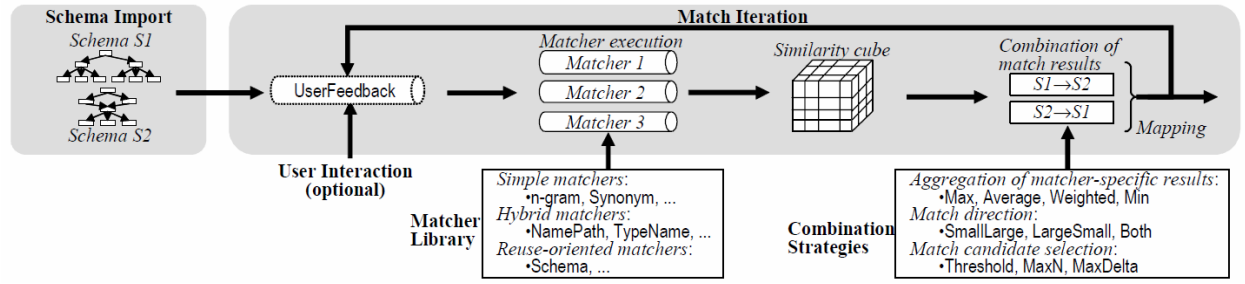

Рис. 2.3. Обробка узгоджень в COMA

COMA використовує алгоритм композитної системи узгодження схем. Використовує синтаксичні та зовнішні техніки. Надає бібліотеку з 6 елементарних, 5 гібридних та 1 алгоритму, орієнтованого на повторне використання. Більшість реалізує техніки на основі рядків (афікс, n-грама, відстань редагування).

Має більш гнучку архітектуру та здатність виконувати ітерації у процесі узгодження порівняно з Cupid. Належить до категорії відображень як підказок подібності.

2.2. Мотивація проблеми узгодження та формалізація відображень

Для мотивації необхідності узгодження та ілюстрації типової ситуації, що виникає в задачі інтеграції даних, розглянемо фрагменти двох каталогів (наприклад, частин Google та Yahoo), представлені на рис. 2.4.

Рис. 2.4. Демонстрація як фрагменти каталогів Google та Yahoo інтегруються у спільну структуру

2.2.1. Процес інтеграції та визначення кандидатів

Першим кроком у процесі інтеграції є ідентифікація кандидатів на узгодження (matching candidates). Наприклад, можна гіпотетично припустити, що:

- Концепт ShoppingO1 є еквівалентним (≡) концепту ShoppingO2.

- Концепт Board GamesO1 є менш загальним (⊆) за концепт GamesO2.

- Індекси (O1,O2) вказують на відповідний вихідний каталог (онтологію).

Після встановлення відповідностей між двома схемами, наступним етапом є генерація виразів запитів, які забезпечують автоматичне перетворення екземплярів даних цих схем відповідно до інтегрованої схеми.

2.2.2. Формалізація елемента відображення (мапінгу)

Ми визначаємо елемент відображення (або мапінг) як 4-кортеж (IDi,j, n1i,n2j,R), де:

IDi,j — унікальний ідентифікатор даного елемента відображення.

n1i — i-й вузол першого графа (i=1,2,…,N1), де N1 — загальна кількість вузлів у першому графі.

n2j — j-й вузол другого графа (j=1,2,…,N2), де N2 — загальна кількість вузлів у другому графі.

R — відношення подібності (семантичний зв'язок) між вузлами n1i та n2j.

У цій роботі ми розглядаємо наступні відношення R, що мають очевидну множинно-теоретичну семантику:

- Еквівалентність: ≡ (однаковість)
- Більша загальність: ⊇ (суперклас)
- Менша загальність: ⊆ (підклас)
- Відношення розбіжності: ⊥ (відсутність спільного значення)

Коли жодне з визначених відношень не виконується, повертається спеціальне відношення Idk (невизначеність).

Узгодження (Alignment) формально визначається як процес виявлення відображень між двома графоподібними структурами шляхом застосування алгоритму узгодження.

2.3. Концептуальна декомпозиція процесу узгодження

Процес узгодження типово структурований двома основними етапами: узгодження на рівні елементів та узгодження на рівні структури [3].

2.3.1. Узгодження на рівні елементів та структури

Алгоритми узгодження на рівні елементів оперують виключно з інформацією на атомарному рівні, що охоплює, наприклад, дані, інкапсульовані в елементах схем. У контексті семантичного узгодження, ці алгоритми продукують семантичні відношення (⊆, ⊇, ⊥, ≡, Idk) замість традиційних коефіцієнтів подібності у діапазоні [0..1]. Ці коефіцієнти подібності часто інтерпретуються як відношення еквівалентності, що характеризуються певним рівнем правдоподібності або впевненості [1].

Алгоритми узгодження на рівні структури зазвичай інтегрують результати, отримані від декількох алгоритмів узгодження на рівні елементів, а також враховують структурні властивості схем.

2.3.2. Узгоджувачі на основі знань

Узгоджувачі на основі знань (УОЗ) використовують як вхідні дані два ідентифікатори концепцій (або синсетів), дефінованих у лексичній базі знань, як-от WordNet. Вони генерують семантичні відношення шляхом експлуатації структурних властивостей цієї бази. У деяких реалізаціях знання, отримані з WordNet, можуть бути скомбіновані зі статистичними даними, агрегованими з великомасштабних текстових корпусів.

Діяльність УОЗ часто базується на мірах подібності або спорідненості. Якщо обчислене значення міри перевищує попередньо визначене порогове значення, формується відповідне семантичне відношення; в іншому випадку повертається відношення Idk (невідомо).

WordNet matcher є відомим представником УОЗ. Він трансформує відношення, надані WordNet, у семантичні відношення згідно з наступною системою правил:

- A⊆B (спеціалізація), якщо A є гіпонімом, меронімом або тропонімом B.

- A⊇B (узагальнення), якщо A є гіперонімом або голонімом B.

- A≡B (еквівалентність), якщо вони пов'язані відношенням синонімії або належать до одного синсету (наприклад, ніч та нічний час).

- A⊥B (контраст/протилежність), якщо вони пов'язані відношенням антонімії або є братами (ко-гіпонімами) в ієрархічній структурі.

Важливо відзначити, що відношення гіпонімії, меронімії, тропонімії, гіперонімії та голонімії характеризуються транзитивністю. Наприклад, виходячи зі структурної ієрархії (рис. 2.5), можна вивести відношення Людина⊆Жива Істота. Якщо жодне з перелічених вище відношень не встановлюється між двома вхідними синсетами, повертається відношення Idk.

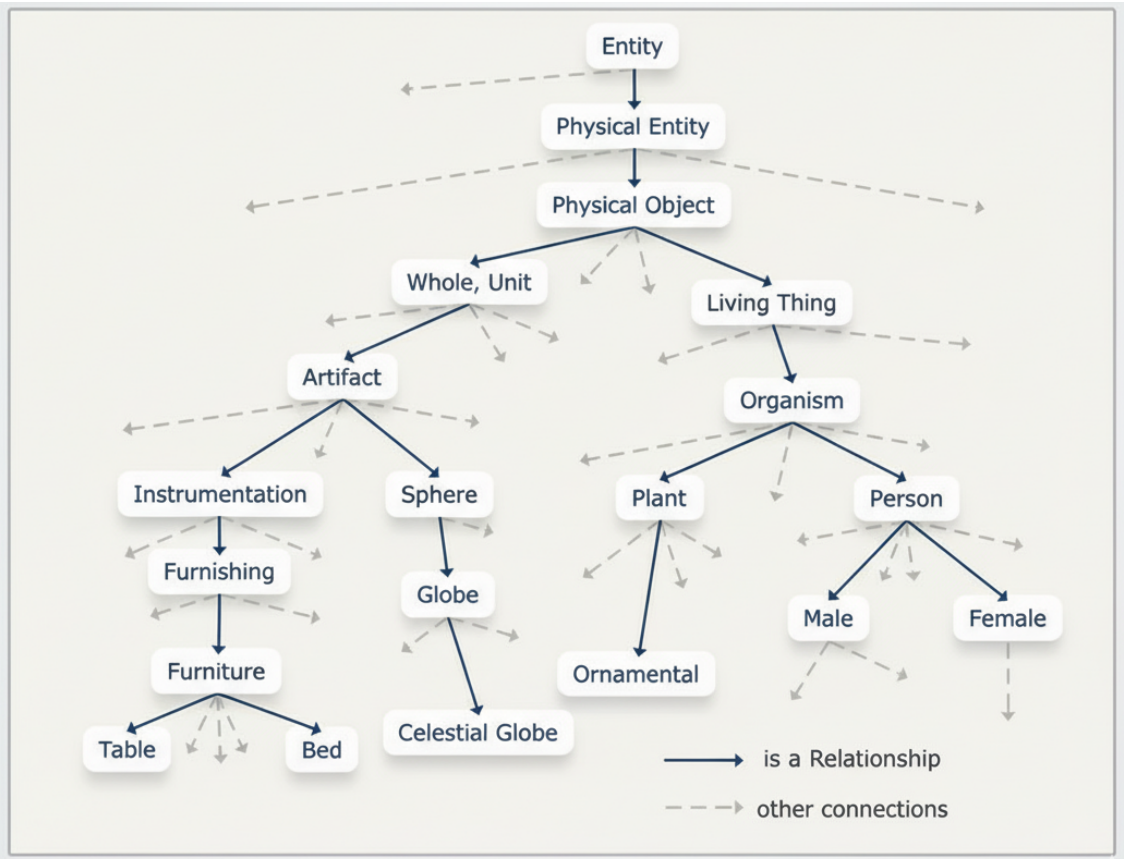

Рис. 2.5. Приклад таксономії засобами WordNet

Наступна таблиця демонструє застосування семантичних відношень до конкретних лексичних пар.

Таблиця 2.1.

Таблиця семантичних відношень у контексті ООП

Вихідний Елемент (Клас A)	Цільовий Елемент (Клас B)	Відношення ООП / Семантичне Відношення	Пояснення / Аналогія ООП
Car	Minivan	⊇ (Узагальнення)	Успадкування (Inheritance): Клас Car є Батьківським (Superclass) для Minivan. Це відношення "is-a" (Minivan — це Car).
Car	Auto	≡ (Еквівалентність)	Синонімічний Концепт / Інтерфейс: Два терміни позначають одну сутність або мають однаковий Контракт.
Tail	Dog	⊆ (Спеціалізація)	Композиція / Агрегація (Composition): Клас Tail є Компонентом (полем) класу Dog. Це відношення "has-a" (Dog має Tail).
Red	Pink	Idk (Невідомо)	Невідоме Відношення / Окремі Класи: Класи не мають прямого успадкування, композиції чи іншого семантичного зв'язку, визначеного в системі.

Наведемо пояснення адаптації:

- Узагальнення (⊇) у випадку car та minivan відповідає Успадкуванню, де загальний клас (Car) є предком, а специфічний (Minivan) — нащадком.

- Еквівалентність (≡) використовується для синонімічних понять (car, auto), які можуть бути просто різними іменами для одного Концепту або Інтерфейсу.

- Спеціалізація (⊆) у випадку tail та dog ідеально підходить для Композиції/Агрегації, оскільки хвіст є частиною собаки, що в ООП реалізується через включення об'єкта як поля класу (відношення "has-a").

- Невідоме (Idk) означає відсутність чіткого відношення успадкування, композиції чи еквівалентності, що є типовим для непов'язаних класів у системі.

Висновки до розділу

У другому розділі досліджено теоретичні та методологічні основи семантичного узгодження природномовного тексту на основі онтологічного підходу. Визначено значення процесу узгодження для забезпечення

сумісності між різними інформаційними системами та джерелами знань. Детально розглянуто підходи до формалізації процесу узгодження, зокрема визначення кандидатів на відповідність і побудову відображень (мапінгів) між елементами онтологій. Проведено класифікацію узгоджувачів на основі знань, що враховують як структурні, так і семантичні ознаки. Показано, що поєднання цих підходів забезпечує вищу точність аналізу семантичних зв’язків у текстах. Таким чином, результати цього розділу стали теоретичною основою для подальшої розробки моделей онтологічного узгодження в експериментальній частині роботи.

РОЗДІЛ 3. ІМПЛЕМЕНТАЦІЯ ОНТОЛОГІЧНИХ МОДЕЛЕЙ АНАЛІЗУ ТА УЗГОДЖЕННЯ СЕМАНТИКИ ПРИРОДНО МОВНОГО ТЕКСТУ

3.1. Засади та застосування в узгодженні онтологій рядкових метрик подібності

3.1.1. Концептуалізація рядкових метрик

Рядкові метрики подібності визначають ступінь схожості між двома текстовими рядками шляхом обчислення їхньої відстані. Ця метрика продукує числове значення, яке кількісно виражає відстань, специфічну для конкретного алгоритму. Завдяки своїй універсальності, рядкові метрики отримали широке використання у сфері інтеграції інформації та низці інших дисциплін.

До ключових областей застосування належать:

- Аналіз даних - дедуплікація даних у базах даних, видобуток даних.

- Інформаційна безпека та криміналістика - виявлення шахрайства, аналіз відбитків пальців.

Біоінформатика - аналіз ДНК, РНК та ідентифікація спільних молекулярних підпослідовностей [19].

- Штучний інтелект (ШІ) та семантичний веб - узгодження онтологій, машинне навчання на основі доказів, інтеграція семантичних знань.

3.1.2. Огляд пов'язаних досліджень та систематизація

Незважаючи на домінування рядкових метрик у розроблених за останнє десятиліття системах узгодження онтологій, існує обмежена кількість систематичних аналізів їхньої ефективності у цій конкретній предметній області [20, 21]. Ці метрики були розроблені та адаптовані в різних наукових галузях: від статистики (для ймовірнісного зв'язування записів [16]) та баз даних (для узгодження записів [17]) до ШІ (для навчання з учителем [18]).

Для класифікації кандидатних узгоджень (candidate mappings) як узгоджених (mapped) або неузгоджених (not mapped) створюються правила узгодження (mapping rules) на основі показників подібності атрибутів (attribute similarity scores). Ці правила визначають, які атрибути або їх комбінації необхідні для точної класифікації кандидатних узгоджень.

На етапі обчислення початкових показників подібності атрибутів невідомо, які саме трансформації (transformations) є доцільними для конкретної прикладної області. Отже, ці початкові показники можуть неточно відображати справжню подібність між об'єктами.

Для коректного обчислення нових показників подібності атрибутів необхідно знати точні ваги трансформацій (transformation weights) для цієї специфічної області. Після класифікації набору кандидатних узгоджень, ваги трансформацій можуть бути визначені шляхом об'єднання інформації як від узгоджень, позначених користувачем, так і від тих, що були класифіковані за допомогою навчача правил узгодження (mapping-rule learner).

Навчач ваг трансформацій розраховує ваги для кожної трансформації, обчислюючи співвідношення між:

- Кількістю разів, коли трансформація була застосована між об'єктами, що узгоджені.

- Загальною кількістю застосувань цієї трансформації.

Отримані ваги згодом використовуються для перерахунку показників подібності атрибутів.

Як показано на рис. 3.1, Навчач Узгодження (Mapping Learner) поєднує два типи навчання — навчання правил узгодження та навчання ваг трансформацій — в одну систему активного навчання. Навчач узгодження поступово навчається класифікувати узгодження між об'єктами, надаючи користувачеві один приклад для позначення за раз. Критерії вибору наступного прикладу для маркування користувачем визначаються вхідними даними від обох навчачів.

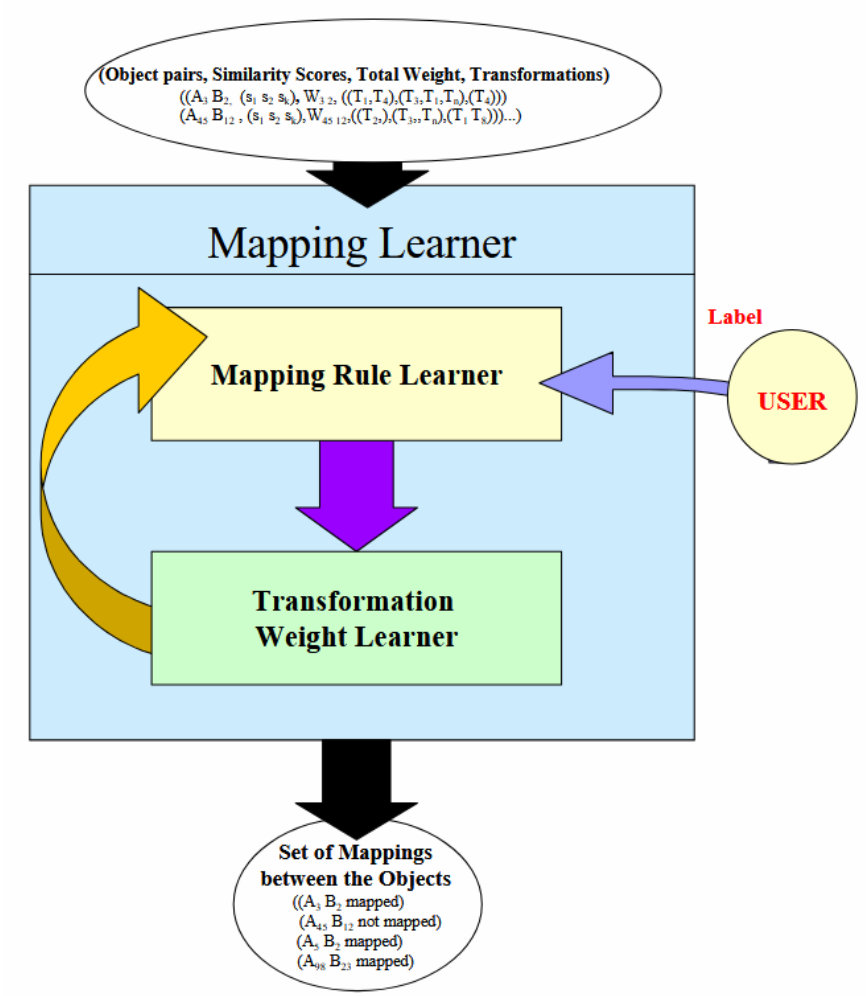

Рис. 3.1. Представлення Mapping Learner

Навчач правил узгодження визначає, який атрибут або комбінація атрибутів (наприклад, Name, Street, Phone) є найбільш важливими для узгодження об'єктів. Метою навчання правил узгодження є досягнення максимально можливої точності в узгодженні об'єктів у різних прикладних областях.

У цьому підході система активно вибирає найбільш інформативні кандидатні узгодження (навчальні приклади), щоб користувач класифікував їх як узгоджені чи неузгоджені. Це робиться для мінімізації кількості позначених користувачем прикладів, необхідних для навчання високоточних правил узгодження.

Навчач правил узгодження складається з комітету навчачів дерев рішень (committee of decision tree learners). Кожен навчач створює власний

набір правил узгодження на основі прикладів, позначених користувачем. Ці правила класифікують приклад. Отримана класифікація використовується:

- Навчачем ваг трансформацій для підвищення точності ваг трансформацій.

- Для прийняття рішення, які саме навчальні приклади мають бути позначені на наступному кроці.

Дослідження, в [20] підкреслили значні розбіжності у продуктивності різних рядкових метрик при застосуванні до тестових наборів Ontology Alignment Evaluation Initiative (OAEI).

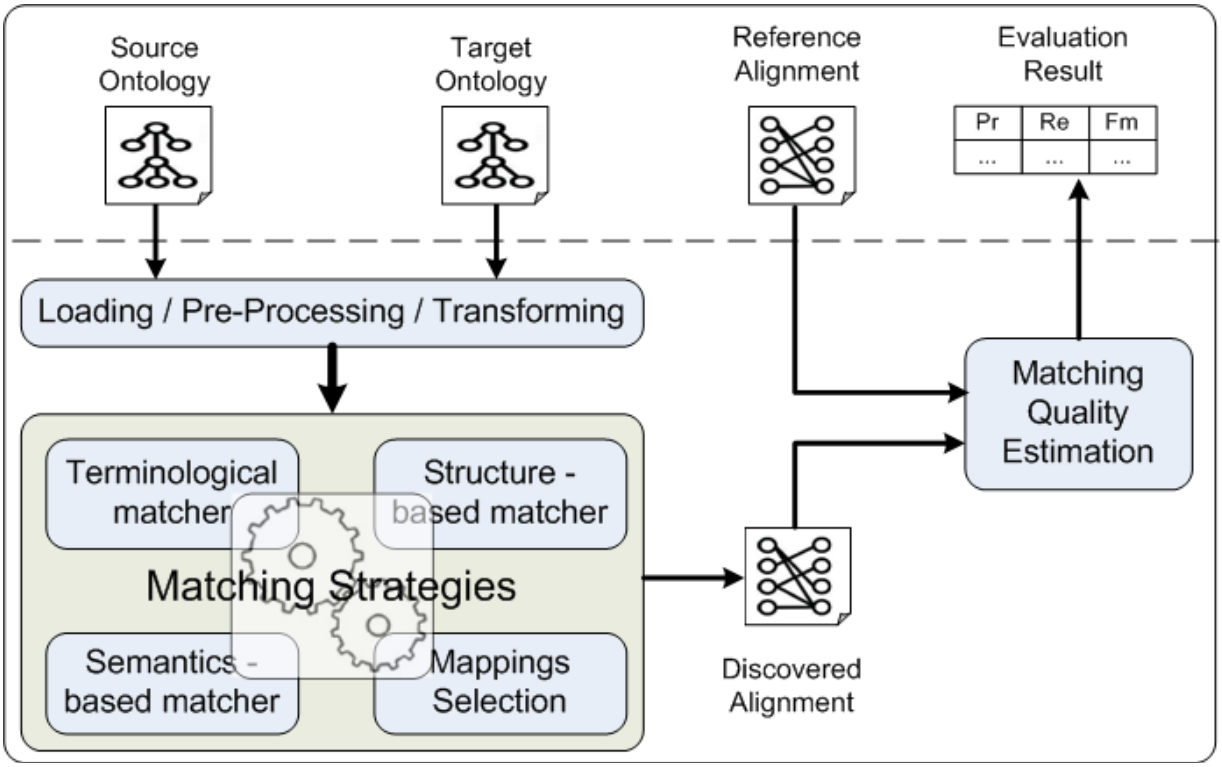

Рис. 3.2. Процес узгодження онтологій

На рисунку 3.2 показано процес узгодження (вирівнювання) онтологій (Ontology Alignment Process), який передбачає виявлення відповідностей між концептами (сутностями) двох різних онтологій.

1. Етап вхідних даних та попередньої обробки (верхня частина)

Процес починається з двох основних вхідних компонентів:

- Source Ontology (вихідна онтологія): онтологія, для якої шукаються відповідники.

- Target Ontology: онтологія, в якій шукаються відповідники.

- Reference Alignment (еталонне узгодження): це попередньо визначений (ручний або відомий) набір коректних відповідностей між двома онтологіями. Він використовується для оцінювання якості знайдених результатів.

Далі ці онтології проходять етап Loading / Pre-Processing / Transforming (завантаження / попередня обробка / трансформація). На цьому етапі дані готуються для узгодження, що може включати токенізацію, нормалізацію рядків, видалення стоп-слів, або інші структурні/синтаксичні модифікації.

2. Етап стратегій узгодження (Matching Strategies)

Це ядро процесу, де використовуються різні методики для пошуку відповідностей:

Terminological Matcher (термінологічний узгоджувач): використовує методи, що базуються на рядкових метриках подібності (наприклад, Левенштейн, Джаро-Вінклер, TF-IDF), для порівняння назв, міток та коментарів сутностей.

Structure-based Matcher (структурний узгоджувач): враховує ієрархічні та графові властивості онтологій, наприклад, порівнює сусідів, предків, нащадків концептів або їхнє розташування у дереві класів.

Semantics-based Matcher (семантичний узгоджувач): застосовує знання, отримані з зовнішніх джерел (наприклад, WordNet) або використовує логічні висновки для визначення семантичних відношень між концептами.

Mappings Selection (вибір узгоджень): після роботи різних узгоджувачів, цей компонент агрегує їхні результати та застосовує фільтрацію або порогові значення для визначення фінального набору виявлених відповідностей.

Результатом цього етапу є Discovered Alignment (виявлене узгодження) - набір відповідностей, знайдений системою.

3. Етап оцінювання (Evaluation)

Якість роботи системи оцінюється шляхом порівняння виявленого узгодження з еталонним - Matching Quality Estimation (оцінка якості узгодження) порівнює Discovered Alignment з Reference Alignment.

Evaluation Result надає кількісні метрики якості, які типово включають:

- Pr (precision / точність): частка правильних узгоджень серед усіх виявлених системою.

- Re (recall / повнота): частка знайдених правильних узгоджень відносно всіх коректних узгоджень в еталоні.

- Fm (F-measure / F-міра): середнє гармонійне між точністю та повнотою, що є інтегральною оцінкою якості.

Рядкові метрики можна систематизувати за трьома основними критеріями.

Таблиця 3.1

Класифікація за обсягом даних (Scope)

Вісь Класифікації	**Категорія Метрики**	**Опис / Принцип**
Обсяг даних	Глобальні Метрики	Потребують інформації про всі рядки в одній або обох онтологіях для узгодження пари; вища часова складність, але більша точність.
Обсяг даних	Локальні Метрики	Потребують лише пару рядків, що розглядаються, як вхідні дані; нижча часова складність.

Таблиця 3.2.

Класифікація за одиницею порівняння (Unit of Comparison)

Вісь Класифікації	**Категорія Метрики**	**Опис / Принцип**
Одиниця порівняння	Метрики на Основі Множин (Set-based)	Визначають подібність шляхом знаходження ступеня перекриття між токенізованими словами, що містяться в рядках. Зазвичай вимагають базової рядкової метрики для порівняння окремих токенів. Ефективні для довгих рядків.
Одиниця порівняння	Метрики, Що Оперують Цілими Рядками (Non-Set-based)	Обчислюють подібність на рівні цілісного рядка або символу, не покладаючись на токенізацію.

Таблиця 3.3.

Класифікація за позицією символу (Sequence Requirement)

Вісь Класифікації	**Категорія Метрики**	**Опис / Принцип**
Позиція символу	Метрики ідеальної послідовності	Вимагають, щоб символи з'являлися в ідентичній позиції в обох рядках для вважання їх відповідними.
Позиція символу	Метрики неідеальної послідовності	Вважають відповідними символи, якщо їх позиції відрізняються лише в межах певного порогу. Краще працюють, коли порядок слів може відрізнятися, але можуть генерувати більше хибнопозитивних результатів.

3.1.3. Рядкові метрики подібності

Рядкові метрики подібності можуть бути категоризовані за трьома основними осями: чи оперують вони множинами слів (токенів), чи вимагають глобальної або локальної інформації, та чи вимагають ідеальної або неідеальної послідовності символів.

1. Метрики на основі множин, глобальні, ідеальної послідовності

Ця категорія метрик базується на множинному перекритті слів, вимагає глобального аналізу онтологій і суворо дотримується ідеальної послідовності (точного збігу токенів).

Ключові Метрики: TF-IDF (Term Frequency / Inverse Document Frequency) та доказовий вміст.

Принцип Дії: Вони зважують спільні слова між двома рядками на основі їхньої рідкісності в усій онтології. Наприклад, TF-IDF вважає сутності більш подібними, якщо вони мають спільне слово, яке є рідкісним у всій сукупності даних.

2. Метрики на основі множин, локальні, ідеальної послідовності

Ці метрики також використовують множинне перекриття слів, але працюють локально (порівнюють лише два рядки) і вимагають точного збігу (ідеальної послідовності) для кожного токена.

Ключові Метрики: джаккард та коефіцієнт перекриття.

Принцип дії: Вони обчислюють ступінь перекриття множин слів. Наприклад, Коефіцієнт Джаккарда визначається як відношення кількості спільних слів до загальної кількості унікальних слів в обох рядках.

3. Метрики, що оперують цілими рядками, локальні, неідеальної послідовності

Ці метрики працюють на рівні цілісного рядка або символу (не множини), оперують локально і допускають неідеальну послідовність, тобто збіг символів або токенів, навіть якщо їх позиції дещо відрізняються.

Ключові метрики: Левенштейн, Джаро-Вінклер, Монж-Елкан, N-грама, Сміт-Ватерман, Стойлос.

Принцип дії: Більшість із них базуються на концепції відстані редагування. Наприклад, Відстань Левенштейна вимірює мінімальну кількість операцій (вставок, видалень, заміщень), необхідних для перетворення одного рядка в інший. Джаро-Вінклер є символьною метрикою, яка надає перевагу рядкам, що мають спільний префікс, а Стойлос – спеціалізована метрика, розроблена для врахування як спільних, так і відмінних рис у контексті узгодження онтологій.

3.1.4. Методи попередньої обробки рядків

Перед обчисленням подібності часто застосовуються методи попередньої обробки для підвищення ефективності узгодження. Вони поділяються на синтаксичні та семантичні підходи.

Синтаксичні методи базуються на символах або мовних правилах, не вимагаючи зовнішніх джерел знань.

Токенізація: Розбиття рядків на слова (токени) за роздільниками або camelCase.

Нормалізація: Усунення стилістичних відмінностей (регістр, пунктуація, нелатинські символи).

Стовбурування/Лематизація: Зведення слів до їхньої основи (наприклад, алгоритм Портера) для усунення граматичних відмінностей.

Видалення стоп-слів: Усунення високочастотних, функціональних слів (наприклад, артиклів, прийменників).

Семантичні методи застосовуються для роботи зі значенням рядків, зазвичай із залученням зовнішніх ресурсів.

Синоніми/Антоніми: Доповнення рядків або врахування протилежності значень через словники/тезауруси.

Переклади: Використання служб перекладу (наприклад, Google Перекладач) для порівняння термінів, написаних різними мовами.

Розширення абревіатур: Трансформація скорочень до повної форми за допомогою зовнішніх знань або правил.

3.2. Специфікація рядкової метрики для узгодження онтологій

Узгодження онтологій є відносно новою галуззю інформатики, внаслідок чого класичні рядкові метрики не були розроблені з урахуванням її специфічних властивостей. Складність алгоритмів узгодження онтологій, що включають численні функції та параметри, такі як поріг (threshold) та кардинальність відображень (наприклад, "один до одного", "один до багатьох"), може суттєво впливати на продуктивність навіть усталених рядкових метрик. Часто ці метрики демонструють незадовільні результати в новому контексті через вплив зазначених параметрів.

З огляду на ці обмеження, формулюються критичні специфікації для розробки нової рядкової метрики, адаптованої до потреб узгодження онтологій

1. Швидкість (Ефективність)

Оскільки онтології застосовуються в додатках, що вимагають обробки в реальному часі (наприклад, у Семантичному Вебі або інтелектуальному пошуку), необхідна низька обчислювальна складність рядкової метрики. Це забезпечить високу швидкість процесу узгодження.

2. Стабільність (Стійкість до Порогу)

Поріг є одним із найважливіших параметрів в алгоритмах вирівнювання. При автоматичній роботі в Семантичному Вебі його оптимальне значення часто фіксується авторами. Хоча існують методи автоматичного коригування порогу [29], їхня здатність щоразу обирати оптимальне значення не є гарантованою. Тому критично важливою вимогою до рядкової метрики є стабільність. Здатність метрики підтримувати майже оптимальну продуктивність навіть за умови невеликих відхилень від оптимального порогу.

Класичні метрики, як правило, дуже чутливі до незначних змін порогу. Їхня висока продуктивність, досягнута за оптимізованого порогу, може швидко знижуватися при невеликій зміні його значення.

3. Семантична точність

У контексті Семантичного Вебу існує ймовірність порівняння онтологій, які є нерелевантними, але містять синтаксично схожі рядки. Метрика повинна бути здатною ідентифікувати всі відмінності та надавати правильні результати, тобто розрізняти концепції з високою подібністю за рядковою ознакою, але різним семантичним змістом.

Проблема полягає в тому, що звичайні рядкові метрики часто не можуть ідентифікувати випадки, коли два рядки представляють абсолютно різні концепції, незважаючи на їхню високу синтаксичну схожість.

Приклад: Пари слів score та store представляють різні семантичні концепції. Проте метрики Монжа-Елкана, Левенштейна, Джаро-Вінклера та інші оцінюють їхню подібність відносно високими значеннями (0.68, 0.8, 0.88 відповідно), що є незадовільним.

4. Уникнення колізій подібності

У випадках, коли вимагається однозначна кардинальність відображень (один до одного), метрика повинна мінімізувати ймовірність присвоєння однакового ступеня подібності при порівнянні одного рядка (з вихідної онтології) з кількома різними рядками (з цільової онтології).

Необхідність виникає якщо рядок відображається з однаковим ступенем подібності на більше ніж один рядок, алгоритм узгодження, ймовірно, не зможе однозначно вибрати правильну пару.

Метрика повинна рідко присвоювати ідентичний ступінь подібності при порівнянні одного конкретного рядка з множиною інших, забезпечуючи високу диференціацію (розрізнення) результатів.

3.3. Застосування метрики відстані Джаро-Вінклера для оцінки подібності рядків під час аналізу тексту

Для підвищення ефективності результатів узгодження рядків застосовується метрика відстані Джаро-Вінклера (Jaro-Winkler distance). Ця міра кількісно оцінює подібність між двома рядками і широко використовується в галузі зв'язування записів (record linkage) для ідентифікації дублікатів.

Чим вище значення метрики Джаро-Вінклера (у діапазоні [0,1]), тим більша подібність між рядками, де 0 відповідає повній відсутності подібності, а 1 — ідентичному збігу. Ця метрика демонструє оптимальну ефективність при порівнянні коротких рядків, таких як власні або особисті назви. Базовим елементом є метрика подібності Джаро (Jaro(s,t)), яка ґрунтується на концепції спільних символів (matching characters) та транспозицій (transpositions).

Ідентифікація спільних символів. Для заданих рядків s=a1...aK та t=b1 ...bL, символ ai в s вважається спільним з t, якщо існує символ bj в t такий, що bj=ai і їхні індекси задовольняють умову вікна:

$$i - H \leq j \leq i + H$$

де H=⌊min(｜s｜,｜t｜)/2⌋ є максимально допустимою відстанню (вікном пошуку).

Нехай s′=a1′...aK′ буде послідовністю символів в s, які є спільними з t (у порядку їх появи в s), і t′=b1′...bL′ буде аналогічною послідовністю в t.

Транспозиція виникає, коли спільні символи ai′ та bj′ знаходяться на однакових позиціях у своїх послідовностях (i=j), але при цьому ai′≠bj′.

Нехай Ts′,t′ — це половина загальної кількості транспозицій між s′ та t′.

Метрика подібності Джаро для рядків s та t визначається як:

$$\text{Jaro}(s,t) = \frac{1}{3} \cdot \left(\frac{|s'|}{|s|} + \frac{|t'|}{|t|} + \frac{|s'| - T_{s',t'}}{|s'|} \right)$$

де: | s | і | t | — довжини рядків s та t.

| s′ | та | t′ | — кількості спільних символів у s та t відповідно (за визначенням, | s′ | = | t′ |).

Ts′,t′ — половина кількості транспозицій.

Варіант Вінклера [16] модифікує метрику Джаро, надаючи перевагу збігу на початку рядка, що особливо ефективно для імен та назв. Модифікація включає врахування довжини найдовшого спільного префікса (P) рядків s та t.

Формула подібності Джаро-Вінклера:

$$\text{Jaro-Winkler}(s,t) = \text{Jaro}(s,t) + \frac{|P|}{10} \cdot (1 - \text{Jaro}(s,t))$$

де | P | — довжина спільного префікса (зазвичай обмежується 4 символами), а 1/10 — коефіцієнт масштабування.

3.4. Методологія вимірювання подібності в узгодженні онтологій

Представлені методології описують два підходи до кількісної оцінки подібності між термінами (t1,t2) або рядками (s1,s2): семантична подібність на основі WordNet та гібридна рядкова метрика.

3.4.1. Вимірювання семантичної подібності на основі WordNet

Для вимірювання семантичної подібності використовується підхід, заснований на рівності значень (senses) у лексичній базі даних WordNet [6].

Критерій рівності: два терміни (t1,t2) вважаються семантично рівними, якщо вони мають принаймні одне спільне значення, яке є синонімом другого терміна.

Функція подібності (SimWN): Міра SimWN(t1,t2) є бінарною. Вона дорівнює 1.0, якщо виконується умова cond(t1,t2), і 0.0 в іншому випадку.

Формальне визначення:

$$\mathrm{Sim}_{\mathrm{WN}}(t_1, t_2) = \begin{cases} 1.0, & \text{якщо } \mathrm{cond}(t_1, t_2) \\ 0.0, & \text{в іншому випадку} \end{cases}$$

де умова cond (t1,t2) визначається як:

$$cond(t1,t2) = \exists x \{x \in senses(t1) \wedge x \in senses(t2)\}$$

Тобто, має існувати хоча б одне спільне значення (x) у множині значень терміна t1 та множині значень терміна t2.

3.4.2. Вимірювання гібридної рядкової подібності

Запропонована гібридна метрика Sim(s1,s2) ґрунтується на інтуїтивній ідеї [17], що подібність між двома сутностями (рядками s1,s2) є функцією їхніх спільних рис (Comm) та відмінностей (Diff). Цей принцип неявно присутній у метриках відстані редагування, де операції редагування рахуються як відмінності, а їх відсутність – як подібність.

Метрика Sim інтегрує компоненти спільності, різниці та покращення за Вінклером:

$$\mathrm{Sim}(s_1, s_2) = \mathrm{Comm}(s_1, s_2) - \mathrm{Diff}(s_1, s_2) + \mathrm{WinklerImpr}(s_1, s_2)$$

де Comm(s1,s2) — функція спільності, Diff(s1,s2) — функція різниці, а WinklerImpr(s1,s2) — покращення результату за методом Вінклера [16].

Функція спільності мотивована розширенням метрики найбільшого спільного підрядка (Longest Common Substring - LCS). Замість одного LCS, вона ідентифікує та агрегує довжини всіх спільних підрядків між s1 та s2 шляхом ітеративного видалення знайденого найбільшого підрядка та пошуку наступного.

У сфері IT дослідники часто використовують описові назви, що поєднують слова (наприклад, numberOfPages і numPages). Ідентифікація кількох спільних підрядків (num, Pages) є критичною для точного відображення реальної подібності та задоволення вимоги розумності метрики.

Формальне визначення:

$$\mathrm{Comm}(s_1, s_2) = \frac{2 \cdot \sum_i \mathrm{length}(\mathrm{maxComSubString}_i)}{\mathrm{length}(s_1) + \mathrm{length}(s_2)}$$

де maxComSubStringi — і-й найбільший спільний підрядок, знайдений ітеративно.

Функція різниці базується на довжинах неузгоджених підрядків, які виникають після початкового етапу узгодження. Припускається, що різниця має відігравати менш важливу роль у загальному обчисленні подібності.

Для агрегування різниці використовується добуток Хамакера (Hamacher product) [30], що є параметричною трикутною нормою.

$$\mathrm{Diff}(s_1, s_2) = \frac{u\mathrm{Len}_1 \cdot u\mathrm{Len}_2}{p + (1-p) \cdot (u\mathrm{Len}_1 + u\mathrm{Len}_2 - u\mathrm{Len}_1 \cdot u\mathrm{Len}_2)}$$

де:

$p \in [0,8)$ є параметром.

uLen1 та uLen2 — це довжини неузгоджених підрядків із s1 та s2 відповідно, масштабовані (нормалізовані) довжиною відповідного рядка.

3.5. Методологія вимірювання семантичної подібності тексту на основі WordNet

3.5.1. Концепція семантичної подібності

Для оцінки семантичної подібності між двома термінами застосовується рівність на основі WordNet (SimWN). Ця міра генерує результат у номінальній формі (бінарній класифікації), де значення 1.0 позначає семантичну рівність ("рівний"), а 0.0 — семантичну нерівність ("нерівний").

Два терміни вважаються семантично рівними, якщо вони мають принаймні один спільний синсет (значення) у базі даних WordNet.

Реалізація цієї методики виконана з використанням мови програмування Python із залученням бібліотеки NLTK (Natural Language Toolkit) для доступу до бази даних WordNet.

Процедура узгодження має наступний алгоритмічний контур:

- Введення: Програма приймає два вхідні терміни (t1 та t2).

- Отримання Синсетів: Використовуючи функціонал WordNet, для кожного вхідного терміна витягуються відповідні синсети (synsets(t1) та synsets(t2)).

- Порівняння: Здійснюється ітераційна перевірка на наявність спільних синсетів між двома множинами.

- Класифікація: Якщо знайдено хоча б один спільний синсет, результатом є семантична рівність (1.0); інакше — семантична нерівність (0.0).

Наведений нижче код реалізує бінарну міру семантичної подібності SimWN (рівність на основі WordNet) шляхом пошуку спільних синсетів між

двома введеними термінами. Для доступу до лексичної бази даних WordNet використовується бібліотека NLTK (Natural Language Toolkit).

Лістинг 3.1. Програмний код для семантичного узгодження на основі WordNet

```
from nltk.corpus import wordnet as wn
import re

# Запит та отримання вхідних термінів від користувача
print('Enter_first_Word')
word1 = input()
print('Enter_2nd_Word')
word2 = input()

# Отримання множин синсетів (значень) для кожного терміна
str1 = wn.synsets(word1)
str2 = wn.synsets(word2)

# Виведення отриманих синсетів для аналізу
print('The_SYNSETS_OF', word1)
print(str1)
print('The_SYNSETS_OF', word2)
print(str2)
print(len(str1))
print(len(str2))

# Ініціалізація змінної для зберігання спільного синсету
x = 'nothing'
# Ініціалізація додаткової змінної (не використовується для фінального результату збігу)
y = 'nothing'

# Перевірка наявності спільних синсетів
for a in str1:
    if a in str2:
        # Знайдено спільний синсет
        x = a
    else:
        # Неспільний синсет
        y = a

# Виведення першого знайденого спільного синсету (або 'nothing', якщо збігів немає)
print('Matches_are:', x)

# Примітка: якщо змінна 'x' містить 'nothing', це відповідає Sim_WN = 0.0.
# Якщо 'x' містить синсет, це відповідає Sim_WN = 1.0.
```

Цей алгоритм реалізує процедуру бінарної семантичної класифікації на основі бази даних WordNet, метою якої є визначення, чи є два вхідні терміни семантично рівними (тобто, чи мають вони принаймні одне спільне значення).

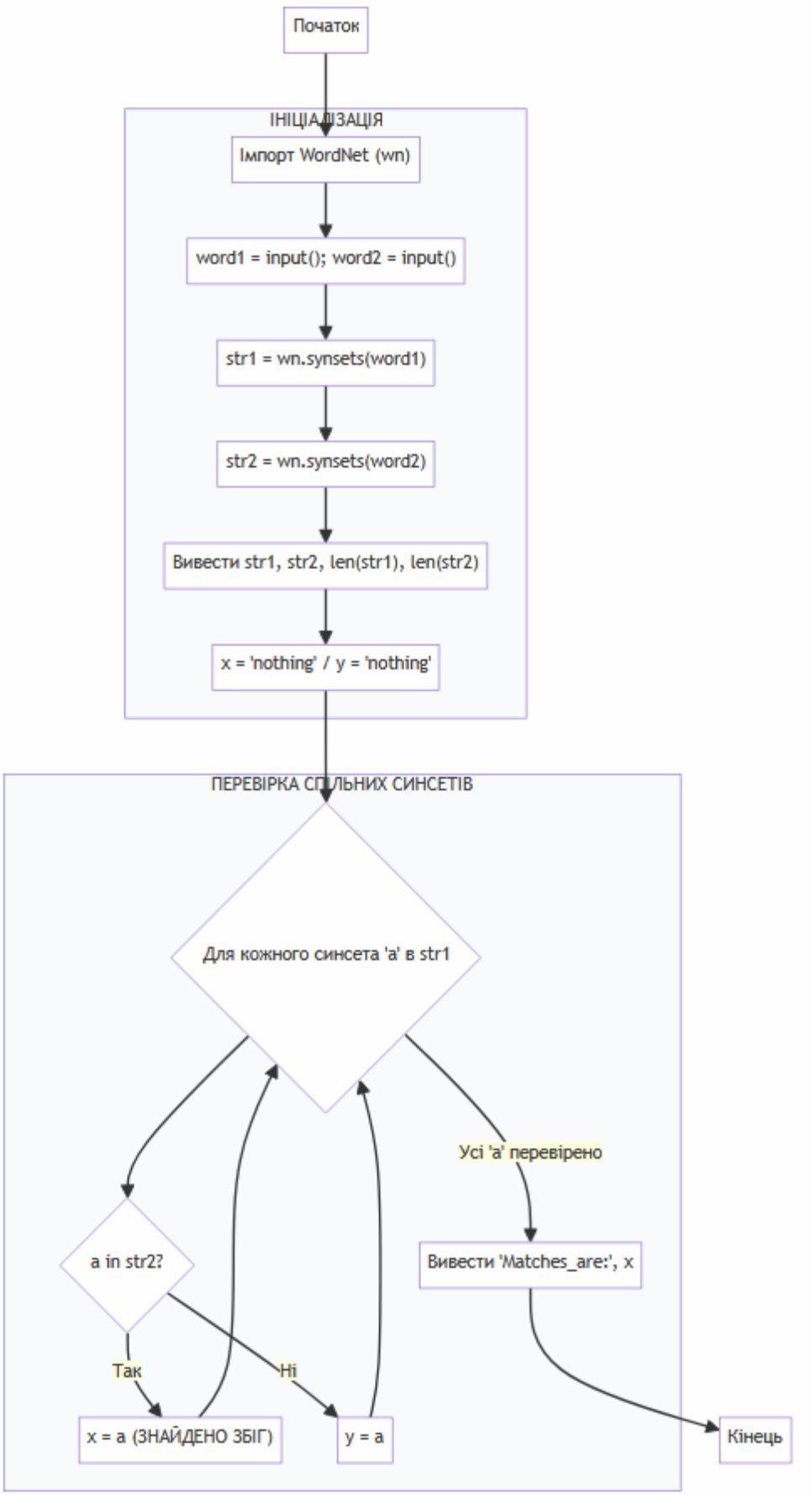

Рис. 3.3. Алгоритмічна схема семантичного узгодження WordNet

Наведемо опис алгоритму семантичного узгодження WordNet:

1. Етап ініціалізації та введення даних

Алгоритм починається з імпорту необхідного функціоналу: модуль wordnet з бібліотеки NLTK для доступу до лексичних даних та модуль re.

1.1. Користувачу пропонується послідовно ввести два слова, які зберігаються у змінних word1 та word2.

1.2. Отримання синсетів: Використовуючи функцію wn.synsets(), для кожного слова витягуються всі його синсети (synsets, які представляють окремі лексичні значення). Результати зберігаються у списках str1 та str2.

1.3. Діагностичне виведення: програма виводить отримані списки синсетів та їхні довжини для обох слів.

1.4. Ініціалізація змінних: ініціалізуються змінні x та y значенням 'nothing'. Змінна x призначена для зберігання першого знайденого спільного синсета.

2. Етап перевірки спільності (бінарна класифікація)

Центральна частина алгоритму виконує перевірку на наявність перетину (спільності) множин значень:

2.1. Алгоритм ітерує (проходить) по кожному синсету a у списку str1 (синсети першого слова).

2.2. У тілі циклу виконується перевірка: if a in str2:. Це є ключовою умовою: чи міститься поточний синсет a у множині синсетів слова str2.

2.3. Фіксація результату:

- Збіг ("рівний"): Якщо спільний синсет знайдено, змінній x присвоюється значення цього синсета (x = a). Оскільки алгоритм шукає хоча б один спільний синсет, змінна x фіксує перший знайдений збіг, і бінарний результат класифікації стає позитивним (відповідає SimWN=1.0).

- Відсутність Збігу: Якщо синсет a не знайдено у str2, виконується гілка else, де y = a. Змінна y не впливає на кінцевий результат подібності.

3. Завершення та виведення

Після завершення циклу алгоритм виводить фінальне значення x під заголовком "Matches_are:". Якщо x містить ім'я синсета, це означає, що семантична рівність встановлена (SimWN=1.0). Якщо x досі містить початкове значення 'nothing', це означає, що спільних значень не знайдено, і терміни вважаються семантично нерівними (SimWN=0.0).

3.5.2. Експериментальна валідація

Експеримент 1. Випадок семантичної рівності (Терміни: "forest" та "wood"):

- forest forest, wood, woods — дерева та інші рослини...; forest, woodland, timberland, timber — земля, покрита деревами...

- wood hard fibrous lignified substance...; forest, woodland, timberland, timber — земля, покрита деревами...; Wood, Natalie Wood...

В результаті аналізу виявлено точний збіг синсетів. Наприклад, значення "forest, woodland, timberland, timber — земля, покрита деревами..." є спільним для обох термінів.

Висновок. Оскільки $\exists x\{x \in \text{senses(forest)} \wedge x \in \text{senses(wood)}\}$, SimWN (forest,wood) дорівнює 1.0. Терміни визнаються семантично рівними.

Експеримент 2. Випадок семантичної нерівності (Терміни: "angry" та "violent"):

- angry (прикметник) feeling or showing anger; (of the elements) as if showing violent anger; severely inflamed and painful.

- violent (прикметник) acting with great force or emotional intensity; caused by force or injury; (of colors or sounds) intensely vivid or loud; marked by extreme intensity of emotion or belief; characterized by violence or bloodshed.

Незважаючи на потенційну конотативну близькість, пряме порівняння множин синсетів у WordNet не виявило жодного спільного елемента.

Висновок. Умова спільності не виконується. Відповідно до застосованого методу, SimWN(angry,violent) дорівнює 0.0. Терміни вважаються семантично нерівними.

3.6. Методика вимірювання гібридної рядкової подібності тексту

3.6.1. Концептуальні засади методики

Вимірювання рядкової подібності базується на визначенні довжин спільного підрядка та неспільного підрядка між двома термінами.

Для отримання остаточної оцінки подібності використовується гібридна метрика, яка інтегрує компоненти спільності (Comm), різниці (Diff) та відстані Джаро-Вінклера (Jaro-Winkler), як це описано у загальному рівнянні:

$$\text{Sim}(s_1, s_2) = \text{Comm}(s_1, s_2) - \text{Diff}(s_1, s_2) + \text{Jaro-Winkler}(s_1, s_2)$$

3.6.2. Імплементація та алгоритмічний опис

Наведений програмний код реалізує обчислення цієї гібридної метрики.

Лістинг 3.2. Код реалізації обчислення гібридної метрики

```
import jellyfish

# Введення вхідних термінів та параметру
print('Введіть перше слово')
word1 = input()
print('Введіть друге слово')
word2 = input()
print('Введіть значення P')
p1 = input()

# Конвертація та визначення довжин
p = float(p1)
len1 = len(word1)
len2 = len(word2)
answer = ""

# 1. Обчислення довжини спільного підрядка (LCS – помилково реалізовано як точний посимвольний
# Примітка: Наведена реалізація фактично знаходить лише спільні символи, що знаходяться
# на ідентичних позиціях (i=j), що не відповідає стандартному визначенню LCS.
for i in range(len1):
    for j in range(len2):
        if (i == j and word1[i] == word2[j]):
            answer = answer + word1[i]

commonlength = len(answer)
```

```
# 2. Обчислення довжин неспільних підрядків
unmatch1 = len1 - commonlength
unmatch2 = len2 - commonlength

# 3. Обчислення компонента спільності (Comm)
# Comm(s1, s2) = 2 * commonlength / (len(s1) + len(s2))
comm = (2 * commonlength) / (len1 + len2)

# 4. Обчислення компонента різниці (Diff) за допомогою добутку Хамакера
# Diff(s1, s2) = (uLens1 * uLens2) / (p + (1 - p) * (uLens1 + uLens2 - uLens1 * uLens2))
# Примітка: У коді не застосовується нормалізація (uLens1, uLens2), як описано в науковому тек
difference = (unmatch1 * unmatch2) / (p + (1 - p) * (unmatch1 + unmatch2 - (unmatch1 * ur

# 5. Обчислення відстані Джаро (використовується як Jaro-WinklerImpr)
jarodistance = jellyfish.jaro_distance(word1, word2)

# 6. Обчислення загальної подібності
result = comm - difference + jarodistance
print(result)
```

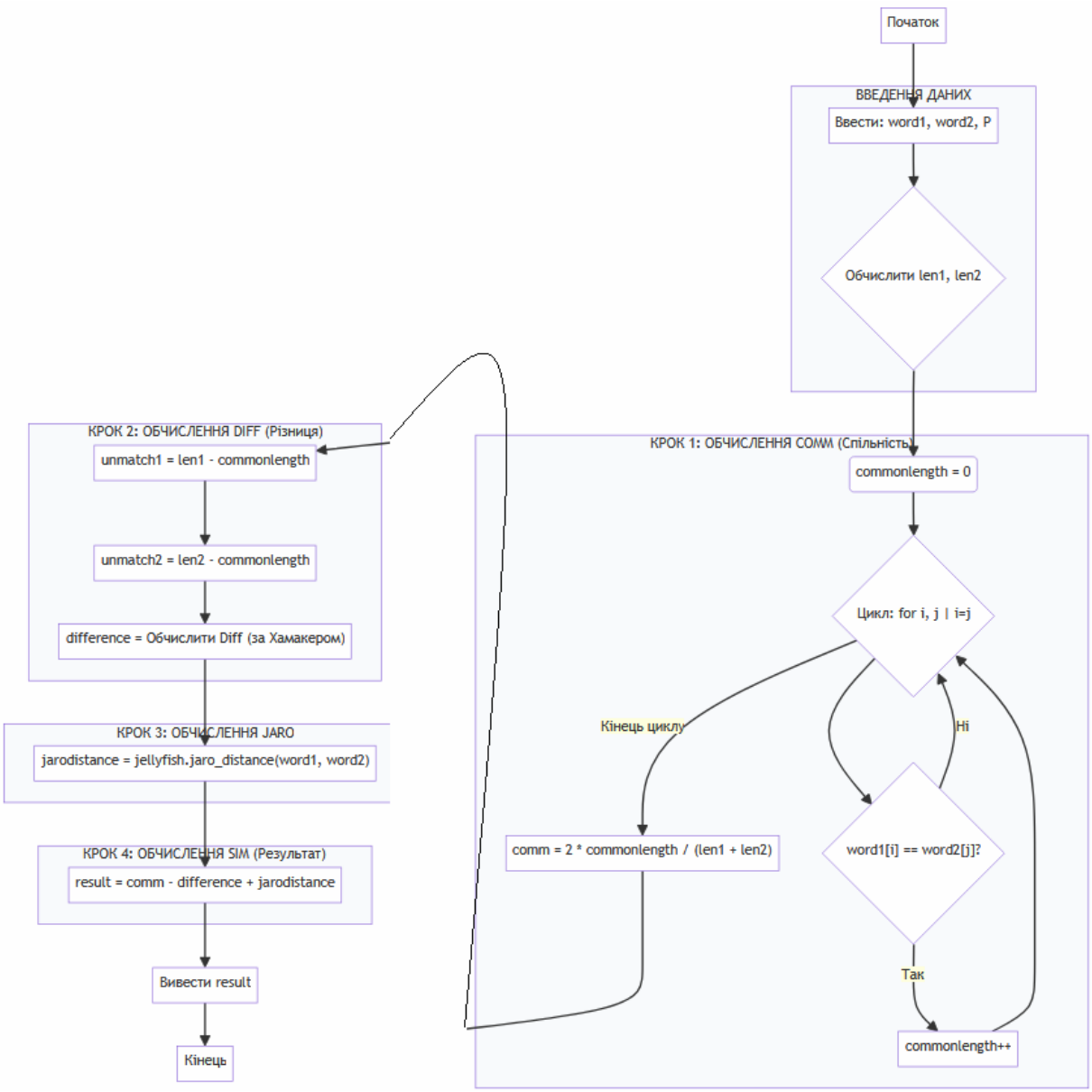

Рис. 3.4. Алгоритмічна схема реалізації обчислення гібридної метрики

Опишемо ключові кроки подані на рис. 3.4:

- Спільність (Comm). На цьому етапі обчислюється Comm, що вимірює ступінь збігу символів. Важливо відзначити, що у наданому коді, реалізація "спільного підрядка" була нестандартною, оскільки враховувала лише символи, що збігаються на ідентичних позиціях (по i=j).

- Різниця (Diff). Компонент Diff обчислюється із застосуванням добутку Хамакера (параметричної t-норми) для агрегації довжин неспільних підрядків (unmatch1, unmatch2). На відміну від теоретичного опису, у коді ці довжини unmatch не були попередньо нормалізовані.

- Jaro-Winkler. Для отримання базової подібності викликається зовнішня функція jaro_distance, яка надає значення, що використовується для покращення загального результату подібності (як WinklerImpr у формулі).

- Фінальний Результат. Гібридна метрика Sim агрегує ці три компоненти. Через віднімання компонента Diff високі значення різниці можуть призводити до значного зниження загальної подібності, потенційно даючи від'ємні значення, як було продемонстровано в експериментах.

3.6.3. Аналіз експериментальних результатів гібридної рядкової метрики

Експериментальне застосування гібридної рядкової метрики, що поєднує спільність (Comm), різницю (Diff) та подібність Джаро-Вінклера (Jaro-Winkler), продемонструвало варіативні результати залежно від морфологічної структури вхідних термінів.

Експеримент 1. "Attribute" та "Attraction"

У випадку порівняння термінів "Attribute" (s1) та "Attraction" (s2) було ідентифіковано чотири спільні символи, що формують підрядок "attr".

Спільність (Comm): Обчислене значення спільності склало 0.421. Цей показник, отриманий як нормалізована довжина спільного підрядка, відображає помірний рівень синтаксичного перекриття між термінами.

Різниця (Diff): Значення різниці, обчислене за добутком Хамакера з використанням параметру p=0.1, досягло 1.744. Це відносно високе значення вказує на значну частку неспільних символів або підрядків.

Jaro-Winkler: Базова подібність Джаро-Вінклера була встановлена на рівні 0.700, що свідчить про високий ступінь схожості, типовий для метрик, що враховують транспозиції та префікси.

Фінальний результат, розрахований за формулою Comm−Diff+Jaro-Winkler, виявився від'ємним, склавши −0.623. Це пояснюється тим, що компонент Різниці (1.744) значно переважає суму компонентів Спільності та Jaro-Winkler, що може бути ознакою чутливості метрики до структурних відмінностей, або ж свідчить про використання ненормалізованих значень неспільних довжин у компоненті Diff.

Експеримент 2. "Price" та "Pride"

При порівнянні термінів "Price" (s1) та "Pride" (s2) було виявлено чотири спільні символи: 'p', 'r', 'i', 'e' (хоча 'e' знаходиться на різних позиціях).

Спільність (Comm): Високе значення спільності становило 0.8, що вказує на суттєве перекриття слів, які мають ідентичну довжину (5 символів) і лише один відмінний символ.

Різниця (Diff): Розрахункове значення різниці становило −1.25 при p=0.1. Слід зазначити, що компонент різниці має бути додатним числом. Ймовірно, це значення вказує на помилку в обчисленні або представленні, де різниця була віднята у формулі, але в тексті експерименту, можливо, вже була представлена зі знаком мінус, який потім віднімається знову. Якщо припустити, що неузгоджена довжина дорівнює 1 для обох слів, очікуване значення Diff мало б бути додатним.

Jaro-Winkler: Значення подібність Джаро-Вінклера також було високим — 0.867, що відповідає дуже незначній синтаксичній відмінності між словами.

Сумарний результат подібності досяг аномально високого значення — 2.917. Це нетипово для метрик подібності, які зазвичай нормалізовані до

діапазону [0,1]. Аномально високий результат є прямим наслідком того, що компонент Різниці був віднятий як від'ємне число (0.8−(−1.25)+0.867), фактично перетворивши його на додатковий внесок до подібності замість штрафу. Цей випадок підкреслює необхідність ретельної нормалізації всіх компонентів та коректної реалізації формули Diff у діапазоні [0,1].

Отже, експериментальне дослідження семантичного узгодження було виконано на основі порівняння синсетів (значень) вхідних термінів. Застосована бінарна метрика SimWN класифікує пару слів як узгоджену ("ЗБІГ") за умови виявлення принаймні одного спільного синсета. Відсутність спільного синсета призводить до класифікації "НЕМАЄ ЗБІГУ".

Ключова перевага даного семантичного підходу полягає в його дискримінативній здатності. Він дозволяє розрізняти терміни, які можуть вважатися подібними за загальновживаним значенням, але мають відмінні або непересічні лексичні значення у структурі WordNet. Таким чином, семантичне узгодження забезпечує більш точну ідентифікацію еквівалентності концептів, ніж винятково синтаксичні методи.

Паралельно було проведено експериментування з гібридною рядковою метрикою, заснованою на інтеграції спільності (Comm), різниці (Diff) та відстані Джаро-Вінклера.

Початкова ідея базувалася на вимірюванні подібності, аналогічному узгодженню підрядків, але з урахуванням як спільних, так і неспільних компонентів. Було встановлено, що базова формула Comm−Diff може генерувати негативні значення, якщо компонент Різниці кількісно переважає компонент Спільності.

Для підвищення точності та розумності результатів, а також для компенсації недосконалості простого порівняння підрядків, до метрики було додано відстань Джаро-Вінклера. Цей елемент функціонує як покращуючий фактор, забезпечуючи більш надійну оцінку подібності, особливо для коротких рядків. В експериментах був обраний фіксований параметр P=0.1 для компонента, що використовує добуток Хамакера (у формулі Diff) та для

модифікатора Вінклера (у Jaro-Winkler), хоча його оптимальне значення може варіюватися залежно від предметної області.

Проведене дослідження було зосереджене на емпіричній оцінці особливостей подібності між парами термінів шляхом реалізації двох основних методик узгодження: семантичного узгодження та рядкового узгодження.

1. Семантичне узгодження

В рамках роботи було частково імплементовано алгоритми семантичного узгодження, які становлять значну частку сучасних систем узгодження онтологій. Семантичний підхід дозволив кількісно оцінити ступінь еквівалентності термінів, виходячи з перетину їхніх лексичних значень (синсетів), що забезпечує дискримінацію між термінами, які є синтаксично схожими, але семантично відмінними.

2. Рядкове узгодження

Для оцінки синтаксичної схожості була реалізована гібридна методика рядкового узгодження. Ця методика базується на вимірюванні спільності та різниці між термінами, а також включає модифікацію за допомогою відстані Джаро-Вінклера. Включення відстані Джаро-Вінклера було необхідним для покращення точності результату, оскільки вона ефективно враховує збіги символів у межах вікна та наявність спільних префіксів, що є важливим для коректної оцінки подібності в термінологічних даних.

Успішна реалізація та обчислення значень для обох типів подібності дозволила продемонструвати, що ступінь схожості між двома термінами може істотно відрізнятися залежно від застосованої методології. Зокрема, було підтверджено, що рядкове узгодження фокусується на морфологічних ознаках, тоді як семантичне узгодження надає більш глибоку оцінку концептуальної еквівалентності. Це підкреслює необхідність використання комбінованих (гібридних) підходів для досягнення високої якості узгодження в задачах обробки природної мови та інтеграції онтологій

Висновки до розділу

У третьому розділі здійснено практичну реалізацію онтологічних моделей аналізу та узгодження семантики природномовного тексту. Визначено роль і можливості застосування рядкових метрик подібності, зокрема відстані Джаро–Вінклера, для оцінки схожості між текстовими одиницями. Розроблено та описано методику вимірювання семантичної подібності на основі лексичної бази знань WordNet, що враховує контекстуальні зв’язки між поняттями. Запропоновано метод гібридної подібності, який поєднує семантичні й рядкові характеристики для досягнення більшої точності узгодження. Проведено експериментальну валідацію, результати якої підтвердили ефективність розробленої методики для аналізу текстів. Таким чином, практична частина роботи довела можливість успішного застосування онтологічних моделей у задачах автоматизованого аналізу та семантичного узгодження природномовних даних.

ВИСНОВКИ

У процесі виконання магістерської роботи на тему «Онтологічні моделі аналізу семантики природно мовного тексту» було проведено комплексне дослідження теоретичних і практичних аспектів побудови, узгодження та застосування онтологічних моделей для аналізу семантики текстових даних природною мовою. Результати дослідження дали змогу сформувати системне бачення процесів концептуалізації знань, семантичного узгодження та їх використання для покращення точності автоматизованих систем обробки текстів.

У першому розділі виконано ґрунтовне дослідження предметної області застосування онтологій для аналізу природномовних текстів. Було визначено, що онтології є ключовим інструментом формального представлення знань, який забезпечує структурування та взаємозв'язок понять у доменній області. Особлива увага приділялася проблемі різноманітності онтологій і потребі в їх узгодженні, що обумовлено існуванням множинних онтологічних моделей, розроблених незалежно одна від одної.

Розглянуто методологічні підходи до узгодження онтологій, серед яких особливо виділено логічні, структурні, лексико-семантичні та гібридні методи. Проаналізовано роль лексичних баз знань, зокрема WordNet, як універсального джерела семантичної інформації, що використовується для автоматизації процесів порівняння та інтеграції онтологій. Дослідження показало, що формалізація концептуалізації домену та побудова ієрархічних структур понять є основою для ефективного моделювання семантичних відношень між термінами природної мови.

У другому розділі розроблено теоретичні та методологічні основи семантичного узгодження природномовного тексту в контексті онтологічного підходу. Визначено значимість узгодження як процесу, що забезпечує інтероперабельність інформаційних систем та уніфікацію змісту

між різними джерелами знань. Було виконано аналітичний огляд сучасних систем і методів узгодження, зокрема схемно-орієнтованих підходів, які забезпечують ефективну інтеграцію даних у гетерогенних середовищах.

Проведено формалізацію процесу відображення (мапінгу) між елементами онтологій, що включає визначення кандидатів на узгодження та побудову відповідних семантичних зв'язків. Розроблено класифікацію узгоджувачів на основі знань і структурних характеристик, що дозволило виділити основні критерії ефективності семантичного зіставлення текстових елементів. У результаті сформульовано загальні принципи побудови моделей узгодження, які поєднують семантичні та структурні ознаки в єдиному концептуальному просторі.

У третьому розділі здійснено практичну імплементацію онтологічних моделей аналізу та узгодження семантики природномовного тексту. Визначено теоретичні засади застосування рядкових метрик подібності для порівняння лексичних одиниць, а також проведено систематизацію існуючих методів вимірювання текстової схожості.

Особливу увагу приділено метриці відстані Джаро–Вінклера, яка була адаптована для задач семантичного аналізу та продемонструвала високу ефективність при порівнянні термінів різного ступеня морфологічної та синтаксичної варіативності. На основі бази знань WordNet розроблено методику вимірювання семантичної подібності, що враховує контекстуальні зв'язки між поняттями, їхню глибину у таксономічній ієрархії та ступінь спільності предків у графі понять.

Крім того, було розроблено методику гібридної рядкової подібності, яка поєднує лексичні та семантичні показники. Запропонована методика була реалізована у вигляді модуля, що забезпечує автоматичну оцінку подібності між текстовими фрагментами на рівні понять і термінів. Експериментальна валідація підтвердила підвищення точності аналізу семантичних зв'язків порівняно з використанням лише окремих метрик.

Отже, результати магістерської роботи мають як теоретичне, так і практичне значення. З теоретичної точки зору, дослідження поглиблює розуміння процесів формалізації семантики природної мови засобами онтологічного моделювання. З практичної — створено інструментальні підходи до реалізації онтологічних моделей аналізу текстів, що можуть бути використані в інтелектуальних системах обробки інформації, аналітичних платформах та прикладних застосуваннях штучного інтелекту.

ПЕРЕЛІК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. J. Euzenat and P. Shvaiko, Ontology Matching, Springer-Verlag Berlin Heidelberg, 2007, pp. 37, 40-42, 65, 92-104.
2. S. Zanobini, Semantic Coordination: The Model and an Application to Schema Matching, PhD Thesis, International Doctorate School in Information and Communicatin Technology, University of Trento, Trento, 2007, pp. 71.
3. Jaap Kamps and Maarten Marx. 2001. Words with attitude. Technical Report PP-2001-16, Institute for Logic, Language and Computation, University of Amsterdam
4. J. Euzenat and P. Valtchev, Similartiy-based Ontology Alignment in OWL-Lite, ECAI European Conference on Artificial Intelligence, Valencia (Spain), 2004, http://www.citeulike.org/user/miguelfm/article/832236.
5. Patrick Blackburn, Maarten de Rijke, and Yde Venema. 2001. Modal Logic, volume 53 of Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, Cambridge UK
6. P. Atzeni, P. Cappellari, and G. Gianforme, “MIDST: Model Independent Schema and Data Translation,” Proc. ACM SIGMOD ’07 (Demonstration), pp. 1134-1136, 2007
7. C. Domshlak, A. Gal, and H. Roitman, “Rank Aggregation for Automatic Schema Matching,” IEEE Trans. Knowledge and Data Eng., vol. 19, no. 4, pp. 538-553, Apr. 2007
8. Learning to map between ontologies on the semantic web | Proceedings of the 11th international conference on World Wide Web - https://dl.acm.org/doi/abs/10.1145/511446.511532
9. Proceedings of the IJCAI-01 Workshop on Ontologies and Information Sharing. - Held on August 4 and 5, 2001 in conjunction with the International Joint Conference on Artificial Intelligence Seattle, USA - https://ceur-ws.org/Vol-47/ONTOL2-Proceedings.pdf

10. S. Melnik, H. Garcia-Molina, and E. Rahm. Similarity Flooding: A Versatile Graph Matching Algorithm. Extended Technical Report, http://dbpubs.stanford.edu/pub/2001-25, 2001
11. Shvaiko, P., & Euzenat, J. (2005). A survey of ontology matching techniques. IEEE Transactions on Knowledge and Data Engineering, 18(1), 84–99.
12. Active Atlas Research Group. (2010-2015). Learning Mapping Rules and Transformation Weights in Active Atlas.
13. Settles, B. (2012). Active Learning. Morgan & Claypool Publishers.
14. Doan, A., Melnik, S., & Alon, H. (2001). Learning soft rules for matching schema and data. International Journal on Very Large Data Bases (VLDB), 10(2-3), 193–210.
15. Rahm, E., & Bernstein, P. A. (2001). A survey of approaches to automatic schema matching. VLDB Journal, 10(4), 334–350.
16. Sabatier, F., & P. J. R. J. B. R. J. Euzenat. (2017). Combining mapping rule learning and transformation discovery for ontology alignment. Journal of Web Semantics, 42, 1–19.
17. Jaro, M. A. (1989). Advances in record linkage methodology as applied to the 1985 census of Tampa, Florida. Journal of the American Statistical Association, 84(406), 414–420.
18. Winkler, W. E. (1990). String comparator metrics and enhanced decision rules in the Fellegi-Sunter model of record linkage. Proceedings of the Section on Survey Research Methods, 354–359.
19. Levenstein, V. I. (1966). Binary codes capable of correcting deletions, insertions, and reversals. Soviet Physics Doklady, 10(8), 707–710.
20. Cohen, W. W., Ravikumar, P., & Fienberg, S. E. (2003). A comparison of string metrics for matching names and addresses. IJCAI Workshop on Information Integration, 3(1), 1–6.
21. Monge, A. E., & Elkan, C. P. (1997). An efficient domain-independent algorithm for detecting approximately duplicate database records.

Proceedings of the 1997 ACM SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery (DMKD), 3–8.

22. Stoilos, G., Stamou, G., & Kollias, S. (2005). Unsupervised Ontology Matching Using Semantic Similarity with Self-Correction. Proceedings of the International Semantic Web Conference (ISWC), 631-645.
23. Miller, G. A. (1995). WordNet: A lexical database for English. Communications of the ACM, 38(11), 39–41.
24. Fellbaum, C. (1998). WordNet: An Electronic Lexical Database. MIT Press.
25. Resnik, P. (1995). Using information content to evaluate semantic similarity in a taxonomy. Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI), 448–453.
26. Li, Y., Bandar, Z. A., & McLean, D. (2003). An approach for measuring semantic similarity between words using multiple information sources. IEEE Transactions on Knowledge and Data Engineering, 15(4), 871–882.
27. Bodenreider, O., & Schopen, M. (2005). Evaluation of a WordNet-based approach to semantic similarity in biomedicine. Proceedings of the AMIA Annual Symposium, 61–65.
28. Budanitsky, A., & Hirst, G. (2006). Evaluating WordNet-based Measures of Semantic Relatedness. Computational Linguistics, 32(1), 13–47
29. Doan, A., Halevy, A. Y., & Ives, Z. G. (2004). Principles of Data Integration. Morgan Kaufmann.
30. Melnik, S., Garcia-Molina, H., & Rahm, E. (2002). Similarity flooding: A versatile graph matching algorithm and its application to schema matching. Proceedings of the 18th International Conference on Data Engineering (ICDE), 117–128.
31. Giunchiglia, F., & Shvaiko, P. (2003). Semantic matching. Proceedings of the 1st European Semantic Web Symposium (ESWS), 397–417.
32. Aumueller, D., Do, H. H., & Rahm, E. (2005). Schema and ontology matching with COMA++. Proceedings of the 2005 ACM SIGMOD international conference on Management of data, 906–908.

33. Fischbach, K., & Gordon, H. (2013). Active learning for ontology mapping: An iterative approach. Journal of Web Semantics, 20, 24–36.
34. Haase, P., Hitzler, P., & Krummenacher, R. (2007). Ontology Engineering with Ontology Networks. IOS Press.
35. Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2002). Item-based collaborative filtering recommendation techniques. Proceedings of the 10th International World Wide Web Conference (WWW), 285–295.
36. Wasserman, S., & Faust, K. (1994). Social Network Analysis: Methods and Applications. Cambridge University Press.
37. Needleman, S. B., & Wunsch, C. D. (1970). A general method applicable to the search for similarities in the amino acid sequence of two proteins. Journal of Molecular Biology, 48(3), 443–453.
38. Manning, C. D., Raghavan, P., & Schütze, H. (2018). Introduction to Information Retrieval. Cambridge University Press.

ДОДАТКИ

Додаток А

Лістинг А.1.

Код що визначає, чи мають два вхідні слова принаймні один спільний синсет (значення) у WordNet

```
from nltk.corpus import wordnet as wn
import re

# Запит та отримання вхідних термінів від користувача
print('Enter_first_Word')
word1 = input()
print('Enter_2nd_Word')
word2 = input()

# Отримання множин синсетів (значень) для кожного терміна
str1 = wn.synsets(word1)
str2 = wn.synsets(word2)

# Виведення отриманих синсетів та їх кількості
print('The_SYNSETS_OF', word1)
print(str1)
print('The_SYNSETS_OF', word2)
print(str2)
print(len(str1))
print(len(str2))

# Ініціалізація змінних для зберігання результату
x = 'nothing'
y = 'nothing'

# Перевірка наявності спільних синсетів
for a in str1:
    if a in str2:
        # Знайдено спільний синсет
        x = a
    else:
        # Неспільний синсет
        y = a

# Виведення першого знайденого спільного синсету (або 'nothing', якщо збігів немає)
print('Matches_are:', x)
```

Лістинг А.2.

Код реалізує обчислення гібридної метрики подібності рядків, що включає компоненти спільності, різниці та відстані Джуро

```
import jellyfish

# Запит та отримання вхідних термінів і параметра P
print('Enter_first_Word')
word1 = input()
print('Enter_second_Word')
word2 = input()
print('Give_the_value_of_P')
p1 = input()

# Конвертація та визначення довжин
p = float(p1)
len1 = len(word1)
len2 = len(word2)
answer = ""

# Обчислення "спільного підрядка" (commonlength)
# Примітка: Цей цикл знаходить лише спільні символи на ідентичних позиціях (i=j)
for i in range(len1):
    for j in range(len2):
        if(i == j and word1[i] == word2[j]):
            answer = answer + word1[i]


# Визначення довжин
commonlength = len(answer)
unmatch1 = len1 - commonlength
unmatch2 = len2 - commonlength

# 1. Обчислення компонента спільності (Comm)
comm = (2 * commonlength) / (len1 + len2)

# 2. Обчислення компонента різниці (Diff) за допомогою добутку Хамакера
difference = (unmatch1 * unmatch2) / (p + (1 - p) * (unmatch1 + unmatch2 - (unmatch

# 3. Обчислення відстані Джаро
jarodistance = jellyfish.jaro_distance(word1, word2)

# 4. Обчислення гібридного результату
result = comm - difference + jarodistance

# Виведення результату
print(result)
```