Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Навчально-науковий центр заочної форми навчання
(повна назва)
Кафедра Інформаційних управляючих систем
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти другий (магістерський)

Дослідження методів автоматизованого поповнення БЗ
на основі повторного використання знань
(тема)

Виконав:
студент _2_ курсу, групи _ІУСТзм-21-1____
Олександра КУРЛУКОВА
(власне ім'я, прізвище)

Спеціальність __122 Комп'ютерні науки
(код і повна назва спеціальності)
Тип програми ____освітньо-професійна
(освітньо-професійна або освітньо-наукова)
Освітня програма _Інформаційні управляючі системи та технології
(повна назва освітньої програми)

Керівник професор каф. ІУС Оксана ЧАЛА
(посада, власне ім'я, прізвище)

Допускається до захисту

Зав. кафедри _____ Костянтин ПЕТРОВ
(підпис) (власне ім'я, прізвище)

2022 р.

Харківський національний університет радіоелектроніки

Навчально-науковий центр заочної форми навчання

Кафедра Інформаційних управляючих систем

Рівень вищої освіти другий (магістерський)

Спеціальність 122 Комп'ютерні науки
(код і повна назва)

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

Освітня програма Інформаційні управляючі системи та технології
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

«_21_» _листопада_ 20 22_ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові Курлуковій Олександрі Юріївні
(прізвище, ім'я, по батькові)

1. Тема роботи ___ Дослідження методів автоматизованого поповнення БЗ на основі повторного використання знань
затверджена наказом університету від _14_ __листопада__2022_ р. № _185Стз_
2. Термін подання студентом роботи до екзаменаційної комісії 16.12.2022 р.
3. Вихідні дані до роботи структура бази знань, удосконалений метод автоматизованого поповнення бази знань; експериментальна перевірка удосконаленого методу
4. Перелік питань, що потрібно опрацювати в роботі Аналіз задачі автоматизованого поповнення БЗ на основі повторного використання знань. Формування рішень для автоматизованого поповнення БЗ. Розробка технології автоматизованого поповнення БЗ на основі повторного використання знань Експериментальна перевірка.

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Аналіз задачі автоматизованого поповнення БЗ на основі повторного використання знань	10.10.22 – 28.10.22	виконано
2	Формування рішень для автоматизованого поповнення БЗ	29.10.22 – 13.11.22	виконано
3	Розробка технології автоматизованого поповнення БЗ на основі повторного використання знань	14.11.22 – 29.11.22	виконано
4	Експериментальна перевірка	30.11.22 – 11.12.22	виконано
5	Оформлення пояснювальної записки та графічного матеріалу	12.12.22 – 24.12.22	виконано
6	Перевірка на плагіат	25.12.22	виконано
7	Захист кваліфікаційної роботи	26.12.22	виконано

Дата видачі завдання _21_ ____листопада____2022 р.

Студент _____  _____
(підпис)

Керівник роботи _____  _____ ____професор кафедри ІУС Оксана Чала
(підпис) (посада, власне ім’я, прізвище)

ЗМІСТ

РЕФЕРАТ

Пояснювальна записка до кваліфікаційної роботи: 69 с., 9 табл., 19 рис., 1 дод., 36 джерел.

БАЗА ЗНАНЬ, МЕТОД АВТОМАТИЗОВАНОЇ ПОБУДОВИ БАЗ ЗНАНЬ, МЕТОД АВТОМАТИЗОВАНОЇ ПОБУДОВИ ПРАВИЛ ДЛЯ БАЗ ЗНАНЬ.

Об’єкт дослідження – процес побудови та поповнення бази знань.

Предмет дослідження – методи автоматизованого поповнення баз знань на основі повторного використання знань.

Метою даної роботи є дослідження методів автоматизованого поповнення БЗ на основі повторного використання знань.

У роботі проведене дослідження методів автоматизованого поповнення БЗ. Проаналізовано існуючі методи побудови БЗ. На підставі проведеного аналізу запропоновано удосконалений метод автоматизованої побудови та поповнення баз знань на основі повторного використання знань.

Галузь застосування задачі, яка розробляється – підприємства будь-якого типу.

ABSTRACT

Explanatory note to the qualification work: 69 pp., 9 tables, 19 figures, 1 appendices, 36 sources.

KNOWLEDGE BASE, METHOD OF AUTOMATED KNOWLEDGE BASE BUILDING, METHOD OF AUTOMATED KNOWLEDGE BASE BUILDING RULES

The object of research is the process of building and replenishing the knowledge base.

The subject of research is the methods of automated replenishment of the knowledge base based on the reuse of knowledge.

The method of this work is the research of methods of automated replenishment of the BZ based on the reuse of knowledge.

In the work, a study of the methods of automated replenishment of the BZ is carried out. Existing methods of BZ construction are analyzed. On the basis of the conducted analysis, an improved method of automated construction and replenishment of knowledge based on the reuse of knowledge is proposed.

The field of application of the task being developed is an enterprise of any type.

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

БД – база даних;

БЗ – база знань;

БП – бізнес процес;

ЗБП – знання-ємні бізнес процеси;

ІДС – інформаційна довідкова система;

ІСПУ – інформаційні системи процесного управління;

ЛММ – логічна мережа Маркова;

ПБЗ – побудова бази знань;

СВЗ – система вилучення знань;

KG – knowledge graph;

NELL – never-Ending learning language;

SQL – structured query language.

ВСТУП

Інформаційні системи процесного управління (ІСПУ) застосовують опис діяльності підприємства як множина бізнес процесів (БП).

Всякий бізнес-процес докладно відтворює послідовність операцій зі створення товарів та продуктів, враховуючи доступні ресурси, користувачів та постачальників, впливів ззовні та не враховують організаційної структури підприємства. Управління підприємством в ІСПУ виконується через управління бізнес-процесами та користуванням їх моделей. Управління бізнес-процесами реалізується враховуючи не тільки параметри результату процесу, а й рівня комфорту клієнта.

Персональні знання частіше за все є неявними, оскільки містять у собі неструктуровані правила та моделі дій. Практичне використання таких неявних знань на підприємстві здебільшого залежить від досвідченості, певного рівня досвіду та мотивування виконавців. В наш час виконувачі в переважній кількості не мають мотивації структурувати свої знання та обмінюватися знаннями. В наші дні методи та інформаційні технології є недостатньо розвиненими, щоб на практиці забезпечити виконавцям плюси при швидкому використанні знань організації підприємства. Тому, ці знання можемо виявити лише внаслідок моніторингу при виконанні ЗБП і не можуть повністю включатися в модель бізнес-процесів при проектуванні. Заповнення бази даних неструктурованою інформацією є давньою проблемою в промисловості та дослідженнях, які охоплюють проблеми вилучення, очищення та інтеграції. Ці проблеми мають на увазі роботу з темними даними та побудова бази знань (ПБЗ). Сучасні підходи та методи побудови баз знань в системах базуються переважно на використанні ідей мікроблогінгу, тобто виникло питання додаткової формалізації знань

У цій роботі ми описуємо DeepDive, систему, яка поєднує ідеї бази даних і машинного навчання, щоб допомогти розробити системи СБЗ та розглядаємо методи для автоматичного поповнення баз знань.

Сучасні бази знань мають і систематизують знання з багатьох різних тематичних областей. Окрім інформації про певну сутність, вони також зберігають інформацію про їхні стосунки між собою.

Оскільки публічні бази знань часто не містять необхідної інформації для аналізу таких сценаріїв виникає потреба у створенні та підтримці спеціалізованих баз знань для конкретної області.

У цій дипломній роботі досліджується метод автоматизованої побудови бази знань на основі аналізу її поведінки, що представлена у вигляді записів про послідовність виконання дій бізнес-процесів.

Об’єкт дослідження – процес побудови та поповнення бази знань

Предмет дослідження – методи автоматизованого поповнення баз знань на основі повторного використання знань.

Метою даної роботи є дослідження методів автоматизованого поповнення БЗ на основі повторного використання знань.

Для досягнення мети, необхідно вирішити наступні задачі:

– аналіз особливостей БЗ;
– дослідження методу автоматизованого поповнення БЗ;
– удосконалення методу автоматизованого поповнення БЗ;
– експериментальна перевірка удосконаленого методу.

1 АНАЛІЗ ОСОБЛИВОСТЕЙ ЗАДАЧІ АВТОМАТИЗОВАНОГО ПОПОВНЕННЯ БЗ

1.1 Аналіз структури та особливостей баз знань

В межах автоматизованого управління базами знань втілюється загальний процес управління всіма знаннями, що потребує розгляду моделей цього процесу управління .

Процес управління знаннями формується з фаз виявлення практичної необхідності у знаннях, визначення джерел для отримання знань, вилучення та охорони знань, а також використання та доповнень знань. Реалізація цих фаз для розв’язання задач підтримки управлінських рішень значною мірою залежить від властивостей знань, що використовуються в організації. Відповідно, методи автоматизованого управління БЗ враховують особливості організаційних знань, та базуються на моделях процесу управління такими знаннями.

Організаційні знання це накопичення певного досвіду, розуміння експертів, важливості цінностей, що забезпечує осередок та основу для оцінювання та включення певної практики та даних».

Виділяють дві базових форми знань в організаційній системі: явні та неявні. Останні підрозділяються на знання за замовчуванням та вбудовані знання. Взаємозв’язок між формами знань представлено на рисунку 1.1.

Явні знання представлені у документарній або в формальній формі. Щоб отримати ці треба виконати пошук знань у сховищах даних та сховищах знань за допомогою спеціалізованої мови запитів, такою мовою може бути SPARQL.

Неявні знання є більш цінними для діяльності організації а ніж з явними.

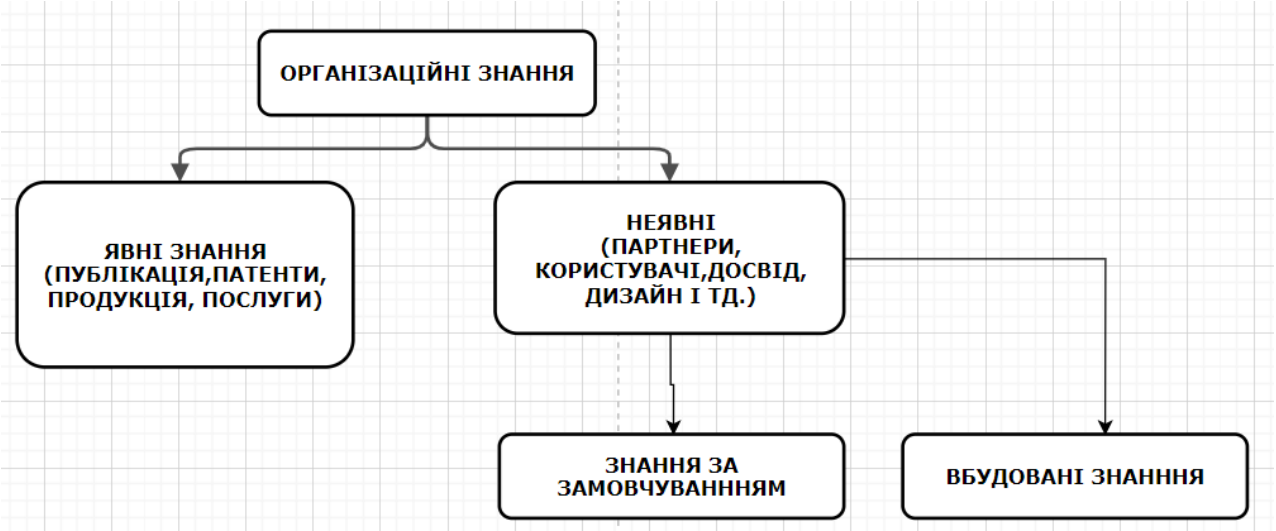

Рисунок 1.1 – Форма представлення знань організації

При управлінні організаційною системою знання постійно змінюються, переходячи с явної форми в неявну та в іншому порядку. Для пошуку та управління знаннями використовуємо базу знань.

Структура баз знань в повній мірі базується на моделі представлення знань, яка зображена на рисунку 1.2.

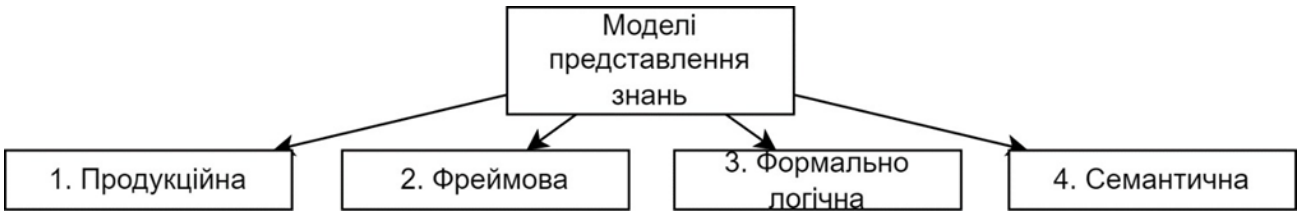

Рисунок 1.2. – Моделі представлення знань

Продукційна модель представлення знань вводить у дію опис знань за допомогою правил. Модель складається з умови та певної дії.

Фреймова модель представлення знань вводить слоти. Слоти в свою чергу є атрибути об’єкта.

Формально-логічна модель представлення знань вводить у дію опис через правила та особлива увага приділяється зв’язкам між сутностями.

Семантична модель описує знання через граф.

Для автоматизованого заселення та поповнення бази знань виконується обробка даних зі зовнішніх джерел. Для цього вхідні дані розбиваємо на елементи. між якими встановлюємо зв’язок. Далі елементи пов’язуємо з існуючими знаннями [3].

Процес побудови БЗ, який описаний та виконаний є автоматизованими. Щоб створити якісну БЗ необхідно обробляти велику кількість темних даних, які повинні періодично повторюватися у різних джерелах, щоб підтвердити факти. Без інструментів автоматизації майже неможливо створити якісну БЗ великого масштабу.

Програмна обробка даних з відструктурованними даними є більш ефективнішою. Методи обробки структурованої інформації стикаються в роботі з меншою кількістю шуму, ніж неструктурованої.

Процес автоматизованої побудови баз знань допомагає швидко підготовлювати інформацію до використання. Це є вкрай необхідним і важливим, адже більшість інформації у інтернеті розташована у неструктурованому вигляді.

Людська мова в форматі тексту зберігається у форматах pdf, word, txt. Наприклад, у медичних закладах карточки пацієнтів зберігаються у формі звичайної мови людей. Мова людей супроводжується багатьма проблемами для програмних систем. Вона є неструктурована і має багато особливостей. Більшість програмних систем є системами, заснованими на правилах, і людська мова вимагає великої кількості правил.

У деяких організаціях, таких як лікарні та інженерні фірми, основну частину даних становлять відео та зображення. Програмне забезпечення для обробки картинок в основному засноване на правилах і може розуміти зображення лише до певного етапу. Але для вилучення повної інформації потрібні ймовірнісні системи.

У центрах гарячої лінії мова зберігається у виді аудіозаписів, що в свою чергу займає велику частину інформації. Мова має велику кількість зайвих

елементів, тобто шуму. Якщо змоделювати мову у виді запису у структуровану форму, то ми значно підвищимо якість роботи організацій та матимемо змогу використовувати ці дані при програмній обробці.

Основна проблема незмодельованих даних криється в тому, що дані можуть зберігатися в необробленому вигляді у БД та не використовуватися взагалі. Політика компанії частіше за все не дають змоги видаляти записи, адже ті містять теоретичну цінність. При необхідності цих записів в майбутньому компанія витратить зайві ресурси для аналізу та вилучення необхідних даних для роботи [4].

Неструктурована інформація погано стискається, що в свою чергу провокує високу дороговизну зберігання невідструктурованих даних, через недостатню обробку та певні витрати на зберігання даних в базі та низьку ефективність обробки інформації.

Проблему поганого стискання може вирішити побудова бази знань. Процес побудови містить в собі вилучення інформації та збереження її у зручному виді, що є вкрай зручним при програмній обробці.

1.2 Дослідження процесу автоматизованого управління БЗ

База знань (БЗ) – це технологія, якою можемо користуватися для пошуку знань та управління ними. БЗ складається зі структурованої інформації при описі частин в предметній області. Це дозволяє виводити інформацію, яка не вказувалася у базі та є основною відмінністю між базою даних та базою знань.

БЗ надають умови для обробки даних, які частіше за все є неструктурованими та працювати з семантикою в програмі.

Сьогоднішні БЗ базуються на основі вилучення знань з неструктурованих даних. Бази знань будуються в процесі вилучення знань.

Процес побудови бази знань виконується автоматизовано або автоматично. Ця особливість дає нам змогу охопити великий об’єм даних.

Завданням побудови бази знань є ідентифікація елементів предметної області та виявлення взаємозв’язків між ними відповідно до обраної моделі представлення знань [6].

Речення в базах знань можуть бути представлені двома компонентами: базою фактів та базою правил. Факти використовуємо для висвітлення основних аксіом в обраній області і є елементарними. Правила є узагальненими домовленістю та застосовуються для збільшення резерву слів, що виражає послідовність побудови свіжих взаємовідносин. Усі факти та висновки з правил можемо одержати через відповідності відносин.

Задача автоматизованої побудови та управління базами знань зводиться до безперервного виконання процесів побудови та застосування та доповнення бази знань, яка представлена на рисунку 1.3.

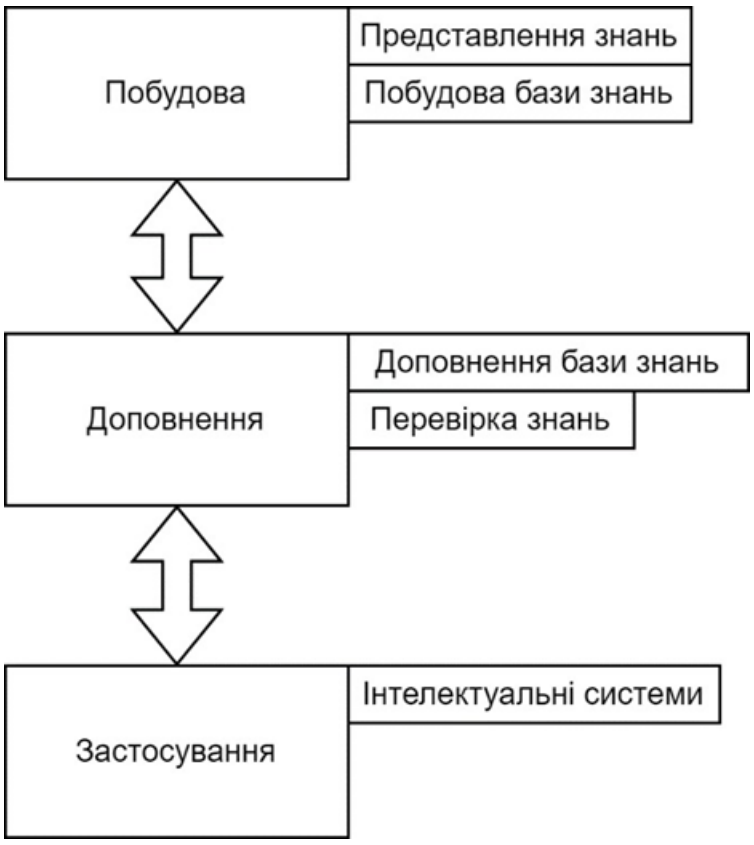

Рисунок 1.3 – Процеси автоматизованого управління базою знань

На етапі побудови бази знань збираємо інформацію з різних джерел та реалізуємо обробка. Виявляємо сутності та їх відповідності.

На етапі доповнення база знань оновлюється новими сутностями та фактами.

Етап застосування бази знань виконує процес висновку над БЗ щоб отримати необхідну інформацію для користувача.

Побудова БЗ – це етап заповнення бази знань інформацією, що була відібрана з неструктурованих джерел. Неструктурованими джерелами вважають текст, відео, відображення, аудіо записи. Неструктуровані дані приводять усю інформацію з неструктурованих джерел, тому маємо певну складність при використанні у програмній обробці. Тобто, база знань є сутністю, яка надає дані у вигляді, де обробка цієї інформації є простою і зрозумілою у порівнянні з неструктурованими джерелами, яким неможливо користуватися під час обробки [22].

Задача побудови БЗ зводиться до виявлення елементів та зв’язків у вибраній предметній області у відповідності з вибраною моделлю.

Забезпечення дієвих управлінських рішень на етапі управління в нових високотехнологічних підприємствах потребує автоматизації задач створення й поповнення бази знань у системах підтримки прийняття рішень. Ці бази включають знання стосовно процесу управління, що дає змогу оперативно та швидко сформувати рішення для вирішення частково структурованих задач. Але концепції, підходи та методи побудови баз знань не зважають на вимоги задля оперативного оновлення знань, які є особливо необхідним та актуальними для нинішніх підприємств, які використовують нові інформаційні технології у діяльності підприємства.

При автоматизованому управлінні базами знань здійснюється спільний процес управління знаннями, якому необхідний розбір і аналіз моделей процесу .

Процес управління знаннями утворюється з етапів визначення практичної необхідності у знаннях, визначення джерел отримання знань,

вилучення та збереження знань, а також застосування знань. Втілення цих етапів при вирішенні задач підтримки управлінських рішень значно залежить від характеристик знань, що використовуються в організації. Відповідним чином, методи автоматизованого управління БЗ враховують специфіку організаційних знань, та основуються на моделях процесу управління цими знаннями.

Згадані знання генеруються на основі даних та інформації про роботу організації. Дані зображують підбірку значень вхідних сигналів (наприклад, у аудіо формі), фактів, які безпосередньо не можуть бути використані. Інформація складається з даних в визначеному зв’язку, які є класифікованими, структурованими, стисненими та зв’язаними із прагненням використання. Інформація дозволяє отримати пояснення на необхідні питання «де, коли, скільки».

1.3 Дослідження методу автоматизованого доповнення БЗ

Щоб знайти рішення для задачі автоматизованої побудови бази знань необхідно скористатися необхідними методами. В таблиці 1.1 було розібрано і представлено можливі методи для побудови бази знань.

Таблиця 1.1 – Методи побудови баз знань

Метод	Опис
Машинне навчання	В методах машинного навчання застосовується напів-закрита система побудови. Вона застосовує зв’язки в великих масивах даних, аби знайти рішення для кінцевого виведення. Вказані методи зручно працюють з великим масивом даних, но в той же час

Кінець таблиці 1.1

	тяжко знаходити помилки, якщо вони зустрічаються, адже система побудови є напів-закритою.
Логічне виведення	Методи логічного виведення баз знань застосовують дану різноманіття правил та фактів щоб дійти логічного висновку при порівнянні сутностей фактів та правил. У цих методах зручно і швидко виявити та змінити похибки, в тому випадку коли вони обмежений набором правил
Статистичне виведення	Методи статистичного виведення застосовують частотні показники фактів й взаємозв'язки у певній множині. Ці методи працюють з великим масивом даних та виявляють найімовірніші факти як найбільш часто зустрічаємо у тексті

Побудова БЗ це складний процес, адже процес містить кілька процесів вилучення даних із джерел. Процес побудови бази знань частіше за все будується з декількох рівнів [23].

Перший рівень містить елементи, що здійснюють вилучення інформації з неструктурованих джерел. Масив усіх неструктурованих даних, з якими працюють елементи першого рівня називають озером даних.

Другий рівень складається з елементів, які роблять аналіз даних, дані в свою чергу збережені в БД в графі першого логічного порядку. Елементи на цьому етапі роблять міркування спираючись на вже збережені дані. Знання у БЗ зображуються у вигляді триплетів. Усі триплети постійно мають по два об'єкта тазв'язок між цими об'єктами.

У базах знань триплети частіше за все йдуть з імовірнісним характером. Ймовірність настання факту збережена для кожного триплету, який описаний. Кожного разу, під час знаходження системою твердження у

новому джерелі, що ідентично пересікається з існуючим фактом, показник імовірності вже присутнього триплету збільшується.

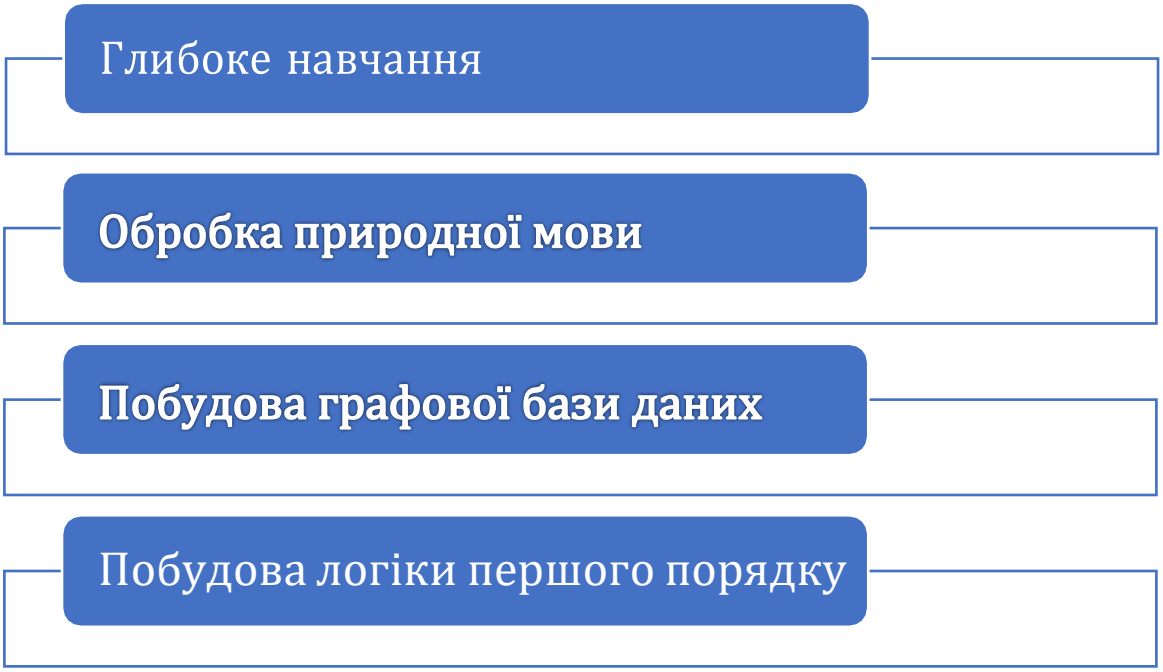

Рисунок 1.4 – Етапи першого рівня

При знаходженні нових доказів, буде підніматися ймовірність фактів, що були знайдені раніше. А також ймовірність новознайдених фактів дуже низька, та буде підійматися з часом при виявленні нових фактів.

Ще в другому рівні ми спостерігаємо системи здобуття знань з БЗ та навчання БЗ. Система навчання БЗ має за ідею накопичення свіжих фактів та доказ старих [25].

Третій рівень має в складі прикладні системи. До прикладних систем входять експертні, пошукові, довідкові, діагностичні.

В наші дні є безліч процесів з встановлення та налагодження автоматизованої побудови баз знань.

Першою серед систем автоматизованої побудови бази знань була Never Ending Language Learning (NELL). NELL – була підготована для утвердження основних семантичних зв’язків між вказаними категоріями даних. Проте в NELL є один мінус, адже тільки малий відсоток інформації є вивірений. В

час комбінування фактів існує імовірність помилки. Факти потребують перевірки експертів.

Google користується Knowledge graph (KG) щоб підняти рівень якості системи пошуку. KG – це сукупність технологій та значень бази знань. KG використовується, щоб доповнювати результат пошуку. Це дозволяє отримувати інформацію без переходу на посилання.

Document360 – це веб-базоване ПЗ щоб керувати знаннями, яке в свою чергу є веб-базованим. Document360 робить по принципу «ПЗ– це сервіс». Платформа впроваджує інструменти для створення, підтримки та використання БЗ для користувачів самообслуговування та внутрішніх клієнтів. Document360 має інструменти:

– текстовий редактор Markdown. Зручний для користувача редактор,який дає змогу використовувати стилі в текстовому документі, при використанні відомих методів форматування, що має в складі заголовки, списки, зображення та посилання;

– менеджер категорій, що дозволяє створювати зміст для навігації по документації. Спрощує користувачам пошук і перетравлення інформації;

– налагодження необхідної сторінки, що допоможе персоніфікувати базу знань завдяки певного дизайну, картинок, зображень стилів, заголовків, включаючи CSS користувачів;

– значення користувачів щоб створити права доступу. Document360 дає змогу використовувати обмеження, хто може отримати можливість використання заданих функцій;

– спосіб щоб контролювати варіанти чи версії, що дозволяє моніторити усі зміни в даному проекті;

Document360 автоматом формує копії усіх проектів кожного дня. Це допомагає відновити проект або його частину БЗ до початкового стану з доступної відкритої копії; допомога інтеграцій, для підключення інших сервісів, або іншої бази знань.

DeepDive – це інформаційно-довідкова система, створена на базі БЗ для вилучення структурованої інформації та даних з темних джерел. Ця система використовує метод машинного навчання. Обробка виконується у DeepDive у певних етапах. Етапи обробки даних зображені на рисунку 1.5.

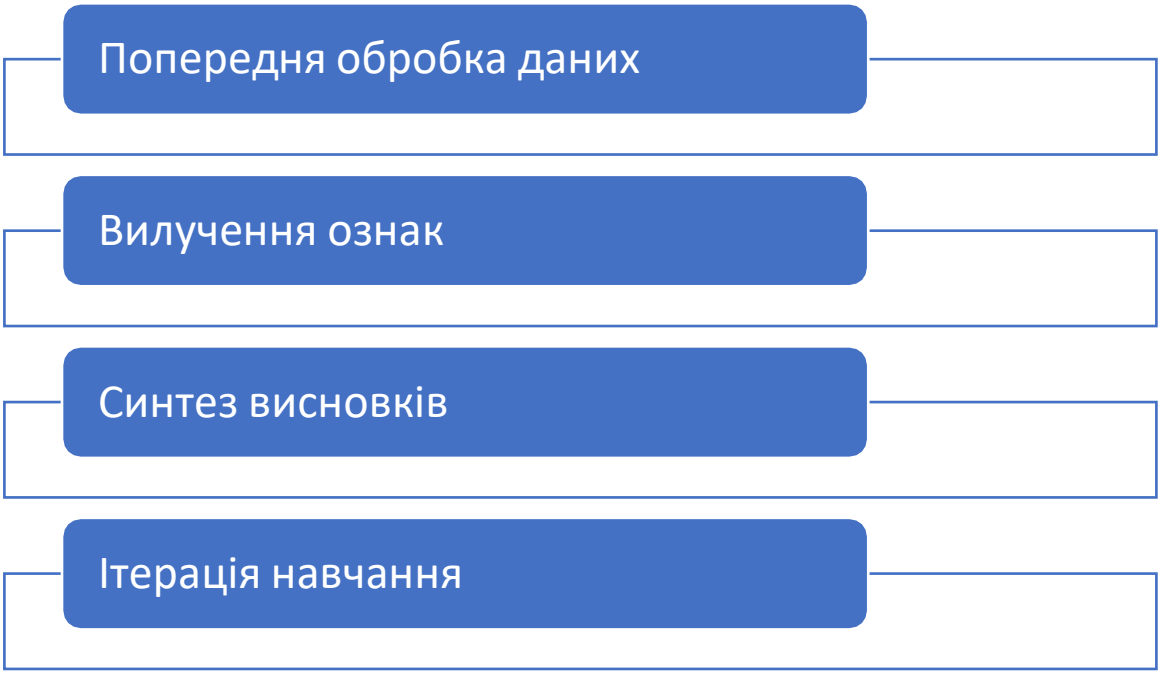

Рисунок 1.5 – Етапи обробки даних в DeepDive

При попередній обробці даних система DeepDive загружає дані на вході у текстовий формат у базу даних та розглядає їх, аби добути інформацію на рівні речення, включаючи слова з кожного речення, теги POS а також теги іменованихоб’єктів та інше.

При вилученні ознак DeepDive перероблює вхідні дані в сигнали відношень, що називають доказами. Докази містять: кандидатів на відносини та певні філологічні риси для цих кандидатів.

Наступний крок DeepDive вставляє підтвердження щоб створити діаграму фактів. Щоб створити факторний граф DeepDive, розробники вживають декларативну мову, дуже схожу до SQL, щоб визначити правила висновку.

Потім DeepDive автоматом робить навчання та генерує статистичний висновок на згенерованному факторному графі. В час навчання обчислюються зміст ваги факторів, котрі прописані в правилах висновку. Ці ваги показують віру до правил. В час висновку підраховуються граничні ймовірності змінних.

Результати зберігаємо в наборі таблиць бази даних, де потім можемо получити результат використавши запит SQL, з'ясувати правильність при допомозі калібрувального графіка та провести аналіз неточностей для підвищення результатів.

На першому кроці заповнюємо БД завдяки набору SQL запитів і функцій відмічених користувачем. DeepDive збирає усі до жодного документа в базу даних [26].

Після кроку заповнення DeepDive здійснює розташування кандидатів через запити SQL, запити утворюють можливі згадування, сутності та відносини, і вилучення ознак, які викликають певну асоціацію функції з кандидатами, наприклад, «(об'єкт) і його дружина (об'єкт)».

Ці правила ми повинні запам'ятовувати, адже якщо зв'язка кандидатів втрачає факт, то DeepDive втрачає шанс вилучити ознаку. Потім нам треба вилучити всі ознаки. Задля цього збільшуємо Марковську модель. По-перше, впроваджуються функції, які визнав користувач. По-друге, вводимо ваги.

Допустимо, що фраза (m1, m2, речення) вертає фразу між двома згадуваннями в реченні, наприклад, «і його дружина» у наведеному вище прикладі. Фраза між двома згадуваннями може вказувати на вірогідність одруження цих людей. Цю фразу записуємо так: ЗгадуванняПроОдруження(m1, m2), що розкладається на ПретендентНаОдруження(m1, m2), Згадування(s, m1), Згадування(s, m2), Речення(s, речення), тоді вага дорівнює фраза (m1, м2, речення) [27].

Вище згадане правило вказує що згадки, вказані в тексті, де згадки m1 і m2 одружені, залежить від речення між парами згадувань. Спираючись на дані про навчання, де згадка що двоє є одруженими, виводить висновок про

впевненість. Ідентифікатор повертається фразою, де ідентифікатор визначає, які коефіцієнти треба використати для вказаного відношення, яке згадувалось в реченні. У випадках коли фраза результує однакове відношення для двох згадок, то вони дістають однакову вагу. Це дозволяє DeepDive підстраховувати прості приклади функцій, наприклад «мішок слів» до словників та онтологій, що є незвичними для домену. На додатку до формулювання наборів класифікаторів, DeepDive наслідує спроможність Марковських моделі відмічати кореляції між сутностями через зважені правила. Такі правила мають особливу користь при очещенні та інтеграції.

Як і в Марковскій моделі, система DeepDive вміє користуватися навчальними даними та доказами про любі відношення; а також, кожне користувальницьке відношення тісно зв’язане з відношенням доказів з однаковою схемою та ще одним полем, що вказує, на істинність чи хибність запису.

На фазі навчання та висновку DeepDive згенеровує факторний граф, який дуже схожий на Марковську модель і реалізовує розширення висновку.

DeepDive виконує всі три фази строго послідовно, і в кінці навчання та висновку отримує крайню ймовірність кожному факту- кандидату. Щоб утворити кінцевий варіант бази даних, користувач обирає факти, де DeepDive однозначно строгий, наприклад, де імовірність більше 0,95. Найчастіше користувачеві потрібно перевірити помилки та повторити перші кроки процесу, який називаємо аналізом помилок. Аналіз помилок – це процес вирішення найуживаніших та найпоширеніших помилок таких, як неправильне виділення, мілко поділені ознаки, помилки користувачів чи кандидатів, тощо.

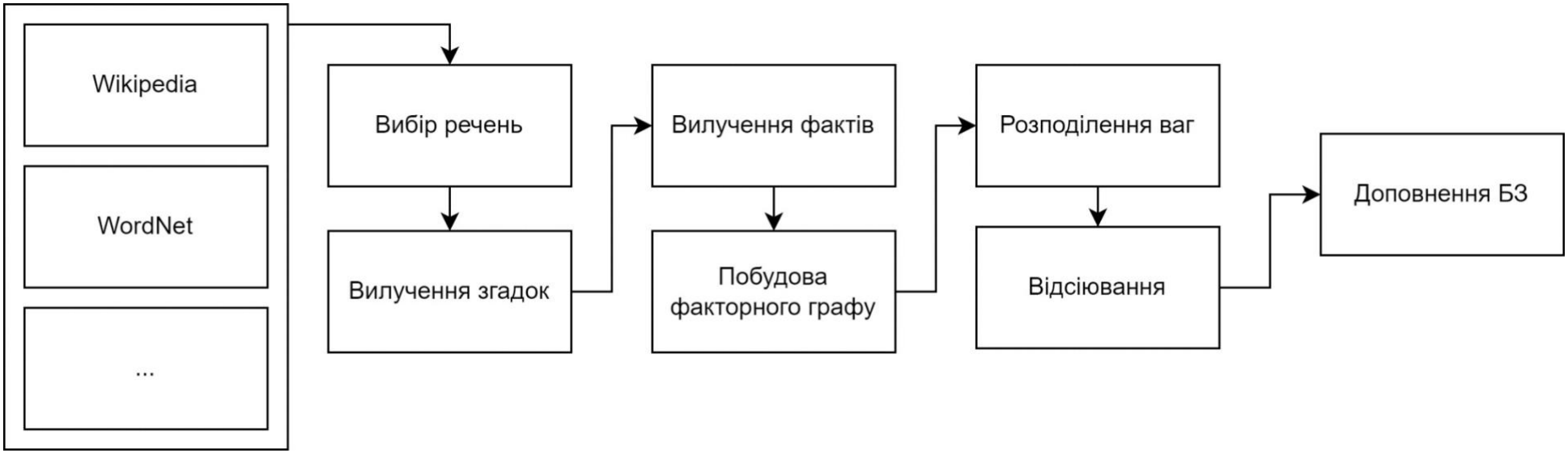

Рисунок 1.6 – Схема процесу обробки даних для представлення їх у базі знань у DeepDive

YAGO достає дані з WordNet та Wikipedia .

WordNet – база англійської мови, яка вивчає смислові значення одиниць мови. WordNet розпізнає слова, які трапляються в текстах, та їх значення. Сукупність слів, що мають однезначення, кличуть синсетом. Тобто, кожен синсет дорівнює одному значенню (або семантичному поняттю). Слова, що приймають кілька значеннь (неоднозначні слова) належать до кількох синсетів.

WordNet дає взаємозв’язки між синсетами, такі як гіпернімія/гіпонімія (тобто зв'язок між підконцептом і суперконцептом) і голонімія/меронімія (тобто відношення між частиною і цілим). Взагалі відношення гіпернімії в WordNet охоплює орієнтований граф,що має один вихідний вузол, що має назву Entity.

Wikipedia – це веб-енциклопедія написана на багатьох мовах. Кожна стаття лише про певну тему та відокремленою сторінкою. Переважна кількість сторінок Вікіпедії власноруч записані в одну або кілька категорій. Сторінка про Володимира Зеленського, наприклад, знаходиться в категоріях політик України, Президент України та ще 24 категорії. У Wikipedia сторінки діляться на категорії та структура посилань відкриті у вигляді таблиць SQL.

YAGO побудована для вилучення онтології з WordNet і Wikipedia. YAGO побудована з перспективою розширення, це означає що до онтології можемо прибавити новітні факти з свіжих джерел. Задля цього факти позначуємо значенням довіри від 0 до 1. Зараз факти зображуються емпіричним рейтингом достовірності, який коливається між 0,90 та 0,98. Факти, які одержані іншими методами (розглянемо приклад:, на основі статистичного навчання), показують замалі значення довіри.

Wikipedia використовує факти про велику кількість людей, ніж WordNet, Інформація про осіб для використання YAGO беруть з Вікіпедії. Назва сторінки «Володимир Зеленський» є кандидатом на те, щоб стати об’єктом. Назви сторінок у Вікіпедії особливі. Щоб утвердити клас для кожної людини, YAGO використовує систему категорій Вікіпедії.

Категорії Вікіпедії структуровані у орієнтований ациклічний граф. Але ця структура зображує тільки тематичну структуру Вікіпедію в сторінках.. Таким чином, ієрархія майже не має користі. Тому YAGO використовує тільки категорії Вікіпедії та ігнорує всі інші категорії. Потім YAGO користується WordNet для зіставлення ієрархії класів, тому що WordNet записує онтологічно чітко окреслену систематику синсетів.

Кожний синсет WordNet стає класом YAGO. YAGO видаляє власні іменники, відомі WordNet, які могли б бути особами.

1.4 Постановка задачі

Ця робота присвячена вирішенню задачі удосконалення методу автоматизованої побудови баз знань на основі повторного використання знань. Актуальність даної задачі є наслідком невідповідності можливостей існуючих методів автоматизованої побудови баз знань для інформаційно-довідкових систем, які використовують додаткові зовнішні бази типових правил, та практичними потребами до процесу автоматизованої побудови баз знань безпосереднього аналізу документів.

Метою роботи є дослідження методів автоматизованої побудови та поповнення баз знань на основі повторного використання знань.

Об’єкт дослідження: процес побудови та поповнення баз знань.

Предмет дослідження: методи автоматизованої побудови баз знань.

У даній магістерській кваліфікаційній роботі вирішуються наступні задачі:

– аналіз особливостей задачі автоматизованого поповнення БЗ;

– аналіз методів представлення знань у базах знань;

– дослідження роботи існуючих методами автоматизованої побудови баз знань;

– аналіз існуючих методів автоматизованої побудови баз знань;

– розробка удосконаленого методу автоматизованої побудови баз знань;

– розробка вимоги до програмної частини;

– розробка програмної частини з використанням удосконаленого методу;

– експериментальна перевірка розробленого методу.

2 АВТОМАТИЗОВАНА ПОБУДОВА ТА ПОПОВНЕННЯ БАЗ ЗНАНЬ

2.1 Методи автоматизованої побудови бази знань

В даній роботі ми розглядаємо метод Probabilistic Knowledge Base, який використовується для автоматизованої побудови бази знань.

На вході маємо дійсну систему з вилучення знань (СВЗ), яка відокремлює з тексту сутності, правила, відношення, класи та факти. З цього скінченного набору констант будується логічна мережа Маркова (ЛММ)

Логічна мережа Маркова – це модель, зображена графом, де множина випадкових величин зображена неорієнтованим графом. За зображенням, мережа Маркова за думкою та зображенням дуже схожа на мережу Баєса, з тою різницею, що граф в мережі Баєса орієнтований та ациклічний, в той час граф ЛММ є неорієнтованим, згідно вище сказаного, може мати цикли. Тобто це математична модель вірогідних фактів та правил, яку можна використовувати при моделюванні імовірнісних БЗ,створених СВЗ.

В ЛММ у кожної формули є свій коефіцієнт, який визначає яка ймовірність настання умови формули.

Приклад логічної мережі Маркова:

$$0.98 \text{ народився в}(\text{Володимир Зеленський}, \text{Дніпропетровськ}), \quad (2.1)$$

$$1.44\ \forall x \in W, \forall y \in P : \text{жив в}(x, y) \leftarrow \text{народився в}(x, y) \quad (2.2)$$

Приклад ЛММ описує факт, що Володимир Зеленський народився в Дніпропетровську, і правило, що якщо політик x народився в області y, то x живе в y. Однак обидва твердження не є однозначними. Ваги 0,98 і 1,44 пояснюють, наскільки твердження правдиві. Обидва твердження є елементом

ЛММ, але кожен має різну мету: твердження є фактом (facts), а правило (rules) описує правило висновку, тому ми аналізуємо кожен поодинці.

ЛММ допускають безкомпромісні правила, які ніколи не змінюються.

Всі правила мають певну вагу ∞ [13].

Наприклад:

$$\infty\, \forall x \in C, \forall y \in C, \forall z \in W :$$
$$(\text{народився в}(z, x) A\ \text{народився в}(z, y) \rightarrow x = y), \qquad (2.3)$$

де x, y, z – факти, C, W – класи.

Написане вище твердження каже, що вибраний об’єкт не може бути народженим в двох різних містах. Факти, що ламають жорсткі правила, розглядаються як помилки і видаляються, аби не зіткнутися з подальшим розповсюдженням.

ЛММ мі розглядаємо як певну заготовку для побудови факторних графів. Де факторний граф зображений як група факторів Ф = {φ1, . . . , φ N }, де кожен фактор φ_i єфункцією $\varphi_i(X_i)$ Знаходиться над стохастичним вектором X_i, та указує на зв'язок серед випадкових величин в X_i. Вказані фактори разом визначають спільний розподіл ймовірностей.

На рисунку 2.1 представлений факторний граф. Квадрати відображають фактори, кола – об’єкти. Зв’язки відображають, які об’єкти який фактор використовує [14].

На рисунку 2.1 у колах відображені цифри, що відповідно:

– народився в (Володимир Зеленський, Дніпропетровськ);
– народився в (Володимир Зеленський, Кривий Ріг);
– жив в (Володимир Зеленський, Дніпропетровськ);
– жив в (Володимир Зеленський, Кривий Ріг);
– знаходиться в (Кривий Ріг, Дніпропетровськ).

А у квадратах відображені фактори відповідно:

– народився в (Володимир Зеленський, Дніпропетровськ);

- народився в (Володимир Зеленський, Кривий Ріг);
- ∀x ∈ W ∀y ∈ P (жив в (x, y) ← народився в (x, y));
- ∀x ∈ W ∀y ∈ P (жив в (x, y) ← народився в (x, y));
- ∀x ∈ P ∀y ∈ C ∀z ∈ W (знаходиться в (x, y) ← жив в (z, x) A жив в (z, y));
- ∀x ∈ P ∀y ∈ C ∀z ∈ W (знаходиться в (x, y) ← народився в (z, x) A народився в (z, y)).

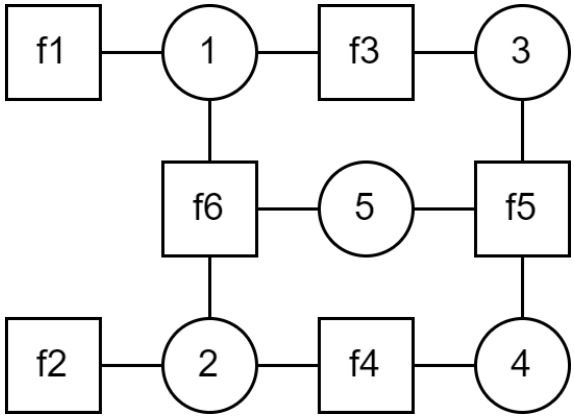

Рисунок 2.1 – Схема факторного графу

Основним факторним графом ми називаємо кінцевий факторний граф

В таблиці 2.1 зображена модель знань для Probabilistic Knowledge Base. Створюємо стохастичний вектор для сутностей та класів, який має одну випадкову величину всіх можливих факторів, що трапляються в правилах та зв'язках. Непередбачені величини, що створюються так, також мають назву основні атоми. Кожен атом приймає значення 0 або 1, що вказує на його правдивість [15].

Імовірнісну БЗ визначаємо наступним способом:

Імовірнісна БЗ представлена у вигляді кортежа з п'яти елементів Г=(E,C,R,П,L), де E– представлений набір сутностей. Кожна з сутностей відповідає реальному об'єкту. C – це набір класів. Усі класи є підмножиною з E. R – сет відношень. Всі відношення означають бінарне відношення на [Ci,

Cj]. [Ci, Cj] називається діапазоном R. R(Ci, Cj) використовується для позначення відношення та діапазону відношення. П – це сет зі зважених фактів. L – це набір зважених правил.

Модель Маркова дозволяє підтримувати спільні формули для першого порядку, проте необхідно провести кордон між правилами та набором правил, які можуть описуваттися диз'юнктом Хорна. В свою чергу диз'юнкт Хорна – це диз'юнкт тільки з однимпозитивним літералом.

Таблиця 2.1 – Модель представлення знань

Сутності (E)	Класи (C)	Зв’язки (R)	Взаємозв’язки (П)
Володимир Зеленський	W (Політик) = { Володимир Зеленський }	народився в(W, P), народився в(W, C)	0,96 народився в(Володимир Зеленський,Дніпропетровськ)
Дніпропетровськ	C (Місто) = {Дніпропетровськ }	жив в(W, P), жив в(W, C)	0,92 народився в(Володимир Зеленський, Кривий Ріг)
Кривий Ріг	P (Місце) = {Кривий Ріг}	знаходиться в(P, C)	
Правила (L)			
1,40 ∀x ∈ W ∀y ∈ P (жив в(x, y) ← народився в(x, y))			
1,53 ∀x ∈ W ∀y ∈ C (жив в(x, y) ← народився в(x, y))			
0,32 ∀x ∈ P ∀y ∈ C ∀z ∈ W (знаходиться в(x, y) ← жив в(z, x) A жив в(z, y))			
0,52 ∀x ∈ P ∀y ∈ C ∀z ∈ W (знаходиться в(x, y) ← народився в(z, x) A народився в(z, y))			
∞ ∀x ∈ C ∀y ∈ C ∀z ∈ W (народився в(z, x) A народився в(z, y) → x = y)			

Його можна записати як:

$$u \leftarrow p, q, \ldots, t, \tag{2.4}$$

де u, p, q, t – факти.

Але це обмежує користування певних правил, але завдяки масштабуванню групі правил системи Шерлок, де отримуємо правила, можна зробити висновок на великому полотні фактів. Диз'юнкти Хорна дають ряд певних плюсів:

Таблиця 2.2 – Базові факти ($T_П$)

I	R	x	C_1	y	C_2	w
1	Народився в	Володимир Зеленський	Політик	Дніпропетровськ	Місто	0.96
2	Народився в	Володимир Зеленський	Політик	Кривий Ріг	Місце	0.93

Таблиця 2.3 – Правила (M_1)

R_1	R_2	C_1	C_2	w
жив в	народився в	Політик	Місце	1.40
жив в	народився в	Політик	Місто	1.53
виріс в	народився в	Політик	Місце	2.68
виріс в	народився в	Політик	Місто	0.74

Таблиця 2.4 – Правила (M_3)

R_1	R_2	R_3	C_1	C_2	C_3	w
знаходиться в	жив в	жив в	Місце	Місто	Політик	0.32
знаходиться в	народився в	народився в	Місце	Місто	Політик	0.52

Алгоритм факторного графу при побудові має два етапи: по-перше, спочатку застосуємо правило для обчислення головних атомів, доки не будемо мати транзитивне замикання. Тоді потрібно використати правило для будови головних факторів [18].

Аби забезпечити правильні фактори потрібно визначити правильні висновки. Задля цього вдаємо, що факти, які ми отримали з джерел,або висновки отримані з фактів з декількох джерел, будуть більш правильні ніж ті що ми зустріли один раз. Потрібно поділити БД для фактів, змішуючи найбільш ймовірні факти в одну таблицю, а менш ймовірні в другу таблицю. Переводимось до переконань,якщо маємо переконання в правильності.

Описаний вище метод розіб’ємо на етапи:

Етап 1 Будуємо факти за рахунок систем вилучення даних аналізуючи тексти.

Крок 1.1 Вибираємо документ,

Крок 1.2 Побудова стеммів (стеммінг).

Крок 1.3 Виділяємо триплети(сутність, властивість, відношення)
Крок 1.4 Будуємо факти спираючись на виділені триплети

Етап 2 Вибір правил, які призначають зв’язок між фактами.

Крок 2.1 Вибираємо правила, які можуть описуватися диз’юнктом Хорна ізсету Sherlok.

Етап 3 Розраховуємо вагу правил

Крок 3.1 Формуємо факторний граф.

Крок 3.2 Розрахунок розподілення за методом Гіббсу

Крок 3.3 Формуємо зважені правила

Крок 3.4 Упорядковуємо правила за вагою

Вище згаданий метод в автоматизованому режимі виконує побудову зваженої бази знань. Використовується набір правил що є статичним та сформован системою Sherlok. На жаль це не дозволяє мати велику кількість фактів, які отримаємо в процесі висновку з основних фактів на основі правил.

Пропонуємо розв’язати проблему, змінивши етап 2, де буде використано вилучення власних правил з джерел, що будуть опрацьовані.

2.2 Удосконалений метод побудови БЗ

Удосконалений метод має на увазі вилучення правил для БЗ із тексту. Для цього робимо припущення, якщо факти зустрінемо в одному джерелі то між ними є зв'язок.

Ми знаходимо де розміщені в множині по два та три факти з усіх джерел. Ці правила записуємо в окрему таблицю. Вага правила більша, коли мі частіше зустрічаємо правило.

Удосконалений метод побудови має наступні етапи:

Етап 1 Будуємо фактів за допомогою систем вилучення інформації.

Крок 1.1 Вибираємо документ

Крок 1.2 Побудова стеммів (стеммінг)

Результатом етапу є підмножина основ

$$Q = \{q_i, p_l\}, \tag{2.5}$$

де Q – це множина слів, p_l – це розділ тексту, q_i – це слово.

Крок 1.3 Виділення триплетів (сутність, властивість, відношення)

$$T = \{(o_i, r_k, s_h): o_i, r_k, s_h \in Q\}, \tag{2.6}$$

де o_i – це об'єкт, r_k – це відношення, s_h – це суб'єкт.

Крок 1.4 Будуємо факти на основі виділених триплетів

$$F = \{f_j: f_j = (o_{j,i}, r_{j,k}, s_{j,h}), \forall(m \neq j), o_j, i \neq o_{m,i} \; or \; r_{j,k} \neq r_{m,k} \; or \; s_{j,h} \neq s_{m,h}\}, \tag{2.7}$$

де o_i – це об'єкт, r_k – це відношення, s_h – це суб'єкт

Етап 2 Будуємо правила, що позначають зв'язки між фактами.

Крок 2.1 Вибір розділу документу (p_l)

Крок 2.2 Вибір фактів в розділі документу

$$F_l = \{ f_j: (o_{j,i}, r_{j,k}, s_{j,h}) \in p_l \}, \quad (2.8)$$

де o_i – це об'єкт, r_k – це відношення, s_h – це суб'єкт

Узагальнене представлення правила можна визначити як:

$$G = \{ g_z(f_j, f_m) \}, \quad (2.9)$$

$$g_z(f_j, f_m) = (s_{j,i}, r_{j,k}, s_{m,h}), \quad (2.10)$$

де f_j, f_m – це факти, r_k – це відношення, s_h – це суб'єкт

Загальна кількість правил тоді буде визначатися як:

$$G = G^{(1)} \cup G^{(2)}, \quad (2.11)$$

де $G^{(1)}$, $G^{(2)}$ – це множини правил першого та другого типів

Крок 2.3 Формування множини розміщень фактів. Множина розміщень по два та три елемента у кортежі буде мати вигляд:

$$A_1 = \{(f_j, f_m) \in p_l \times p_l\}, \quad (2.12)$$

$$A_2 = \{(f_j, f_m) \in p_l \times p_l\}, \quad (2.13)$$

$$A_2 = \{(f_j, f_m, f_n) \in p_l \times p_l\}. \quad (2.14)$$

де f_j, f_m, f_n є фактами, q_i є множиною усіх фактів з розділу документу.

Етап 2. Доповнення темпоральної бази знань.

Крок 2.1. Доповнення бази знань підмножиною незважених темпоральних правил.

Крок 2.2. Уточнення ваг темпоральних правил у базі знань.

Етап 3. Аналіз поточного стану об'єкту управління, виявлення аномального стану у випадку його виникнення.

Крок 3.1. Обчислення ваги темпоральних правил із множини, відображають знання про досягнення поточного стану ОУ.

Крок 3.2. Відбір підмножин темпоральних правил, що містять знання про досягнення поточного стану на послідовностях.

Крок 3.3. Виявлення аномального стану на основі порівняння оцінок темпоральних правил, отриманих в результатів виконання кроків 3.1. та 3.2.

Етап 4. Формування, у випадку виявлення аномального стану об'єкту управління, нового багатоваріантного управлінського рішення, що містить послідовності станів (управляючих дій) від поточного до цільового стану ОУ.

Розроблений метод об’єднує декілька задач поповнення бази знань, так формування управлінського рішення. Метод спрямований на реалізацію при виникненні нових станів ОУ, що дає змогу побудувати нові послідовності дій використовуючи актуальний набір зважених темпоральних правил.

3 РОЗРОБКА ТЕХНОЛОГІЇ АВТОМАТИЗОВАНОЇ ПОБУДОВИ ТА ПОПОВНЕННЯ БЗ НА ОСНОВІ ПОВТОРНОГО ВИКОРИСТАННЯ ЗНАНЬ

Для автоматизованої побудови баз знань у системах пропонуємо технологія, схема функціональної діаграми якої представлена на рисунку 3.3.

Інформацію з документів та інтернету експлуатуємо як вхідну інформацію, яка відображає сутності та зв’язки між ними.

На першому етапі виконуємо вилучення знань з інтернет-джерел. Існуючі системи автоматом достають сутності, факти та правила з Інтернету. Через великий рівень шуму велика кількість із цих витягів є невизначеними.

На другому етапі використовуємо удосконалений метод.

Припустимо, що отримані кортежі є правилами з малою вагою.. Кожен тип правила і має таблицю Rules, в якій записуються предикати, залучені до правил цього типу. Для кожного правила ми маємо кортеж у Rules.

У таблицях пишемо тільки предикати. Порядок задається правилами.

Великі правила можуть викликати певні протиріччя, оскільки кількість шаблонів правил зростає з розміром правила.

Далі виконується класифікація сутностей.

Будуємо факторний граф. Де факти та правила записуємо в базу даних. Завдяки зробленій реляційній моделі побудова факторного графу виконуємо декілька операцій JOIN на таблицях фактів. По-перше, виконуємо запит для виводу фактів. Запит виконуємо циклічно, поки не кінчяться факти. Приклад запиту дивіться на рисунках 3.1 та 3.2.

Далі вже інший запит генерує факти. Він зображений на рисунку 3.3.

Запит на рисунку 3.3 повертає факторну матрицю. Виконується поєднання таблиць правил та фактів та будується матриця зв’язків між правиламита фактами.

```
SELECT Rules2.Relationship1 AS R,
Relationships.Entity1 AS x, Relationships.Class1 AS C1
Relationships.Entity2 AS y, Relationships.Class2 AS C2
FROM Rules2
JOIN Relationships ON Rules2.Relationship2 = Relationships.Relation
AND Rules2.Class1 = Relationships.Class1
AND Rules2.Class2 = Relationships.Class2
```

Рисунок 3.1 – Запит для виводу фактів

```
SELECT Rules3.Relationship1 AS R,
Relationships2.Entity2 AS x, Relationships2.Class2 AS C1,
Relationships3.Entity2 AS y, Relationships3.Class2 AS C2
FROM Rules3
JOIN Relationships Relationships2
ON Rules3.Relationship2 = Relationship2.Relation AND
Rules3.Class3 = Relationships2.Class1
AND Rules3.Class1 = Relationships2.Class2
JOIN Relationships Relationships3
ON M3.Relationship3 = Relationships3.Relation
AND Rules3.Class3 = Relationships3.Class1
AND Rules3.Class2 = Relationships3.Class2
WHERE Relationships2.x = Relationships3.x
```

Рисунок 3.2 – Запит для виводу фактів

На третьому етапі виконується відсіювання фактів.

```
SELECT Relationships1.idRelationships AS I1,
Relationships2.idRelationships AS I2,
Relationships3.idRelationships AS I3,
Rules3.w AS w
FROM Rules3
JOIN Relationships Relationships1 ON
Rules3.Relationship1 = Relationships1.Relation
AND Rules3.Class1 = Relationships1.Class1
AND Rules3.Class2 = Relationships1.Class2
JOIN Relationships Relationships2 ON
Rules3.Relationship2 = Relationships2.Relation
AND Rules3.Class3 = Relationships2.Class1
AND Rules3.Class1 = Relationships2.Class2
JOIN Relationships Relationships3
ON Rules3.Relationship3 = Relationships3.Relation
AND Rules3.Class3 = Relationships3.Class1
AND Rules3.Class2 = Relationships3.Class2
WHERE Relationships1.x = Relationships2.y
AND Relationships1.y = Relationships3.y
AND Relationships2.x = Relationships3.x;
```

Рисунок 3.3 – Запит для генерації факторів

На четвертому етапі виконуємо вибіркову перевірку експертом. Для цього вибираються випадкові факти для ручної перевірки. При правильності фактіві, нічого не змінюється. Якщо виявлена помилка, помилковий факт виділяється та виконується побудова факторного графу на основі фактів, Ваги цих фактів можемо корегувати або вилучати факти.

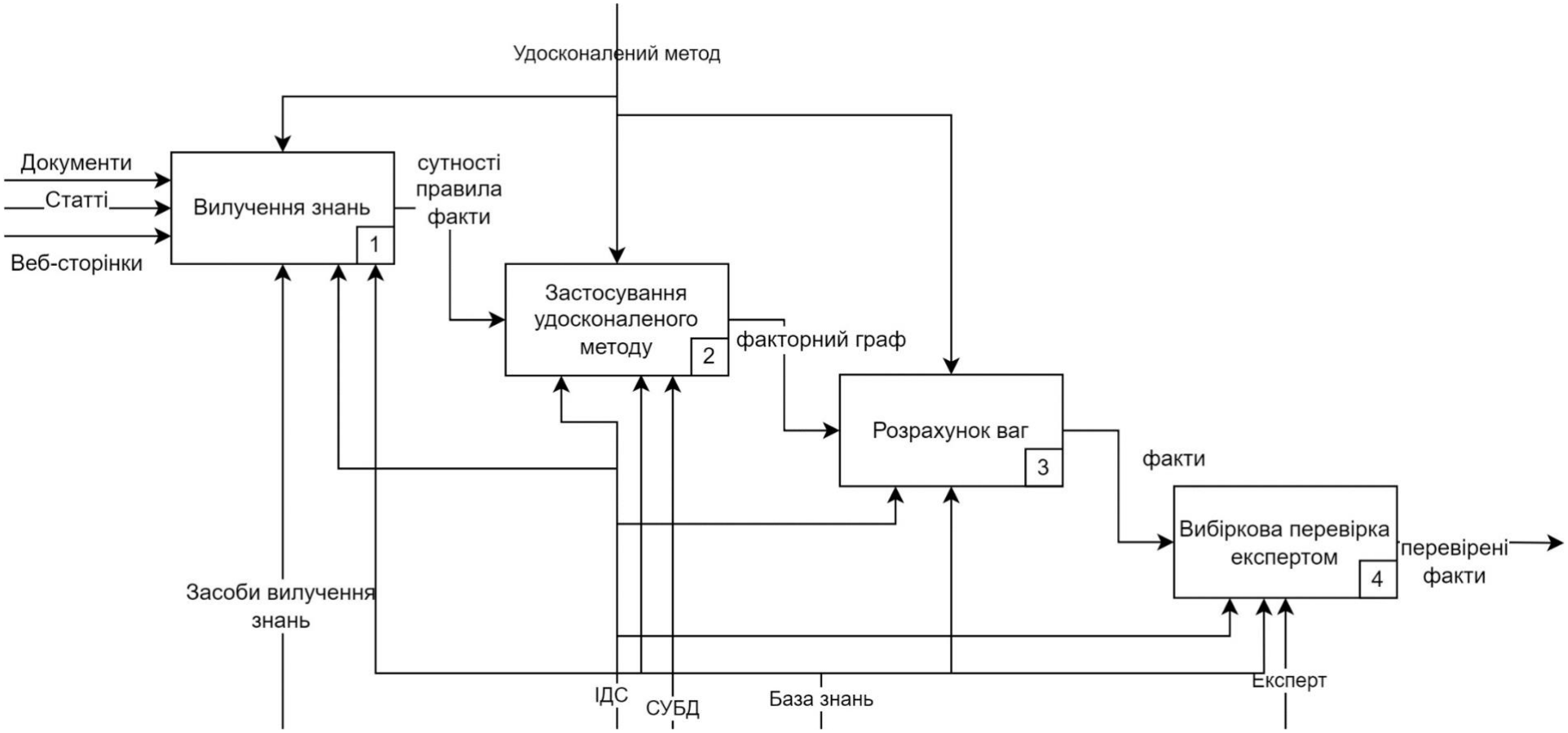

Рисунок 3.4 – Схема функціональної діаграми технології для автоматизованої побудови баз знань

4 ПРАКТИЧНЕ ВИКОРИСТАННЯ ОТРИМАНИХ РЕЗУЛЬТАТІВ

4.1 Розробка програмного засобу автоматизованої побудови зваженихправил для баз знань в інформаційно-довідкових системах

Таблиця 4.1 – Базові факти

I	R	x	C_1	y	C_2	w
1	народився в	Володимир Зеленський	Політик	Дніпропетровськ	Місто	0.96
2	народився в	Володимир Зеленський	Політик	Кривий Ріг	Місце	0.93

Таблиця 4.2 – Правила першого типу

R_1	R_2	C_1	C_2	w
жив в	народився в	Політик	Місце	1.40
жив в	народився в	Політик	Місто	1.53
виріс в	народився в	Політик	Місце	2.68
виріс в	народився в	Політик	Місто	0.74

Таблиця 4.3 – Правила третього типу

R_1	R_2	R_3	C_1	C_2	C_3	w
знаходиться в	жив в	жив в	Місце	Місто	Політик	0.32
знаходиться в	народився в	народився в	Місце	Місто	Політик	0.52

Для описаного методу потрібно зобразити базу знань як реляційну модель. Це дозволить використовувати функціонал, а також реалізований СУБД, при використанні мови запитів SQL

Таблиця Entities містить у собі інформацію про сутності.

Таблиця Classes містить інформацію про класи в системі.

Таблиця Relations містить інформацію про можливі відношення.

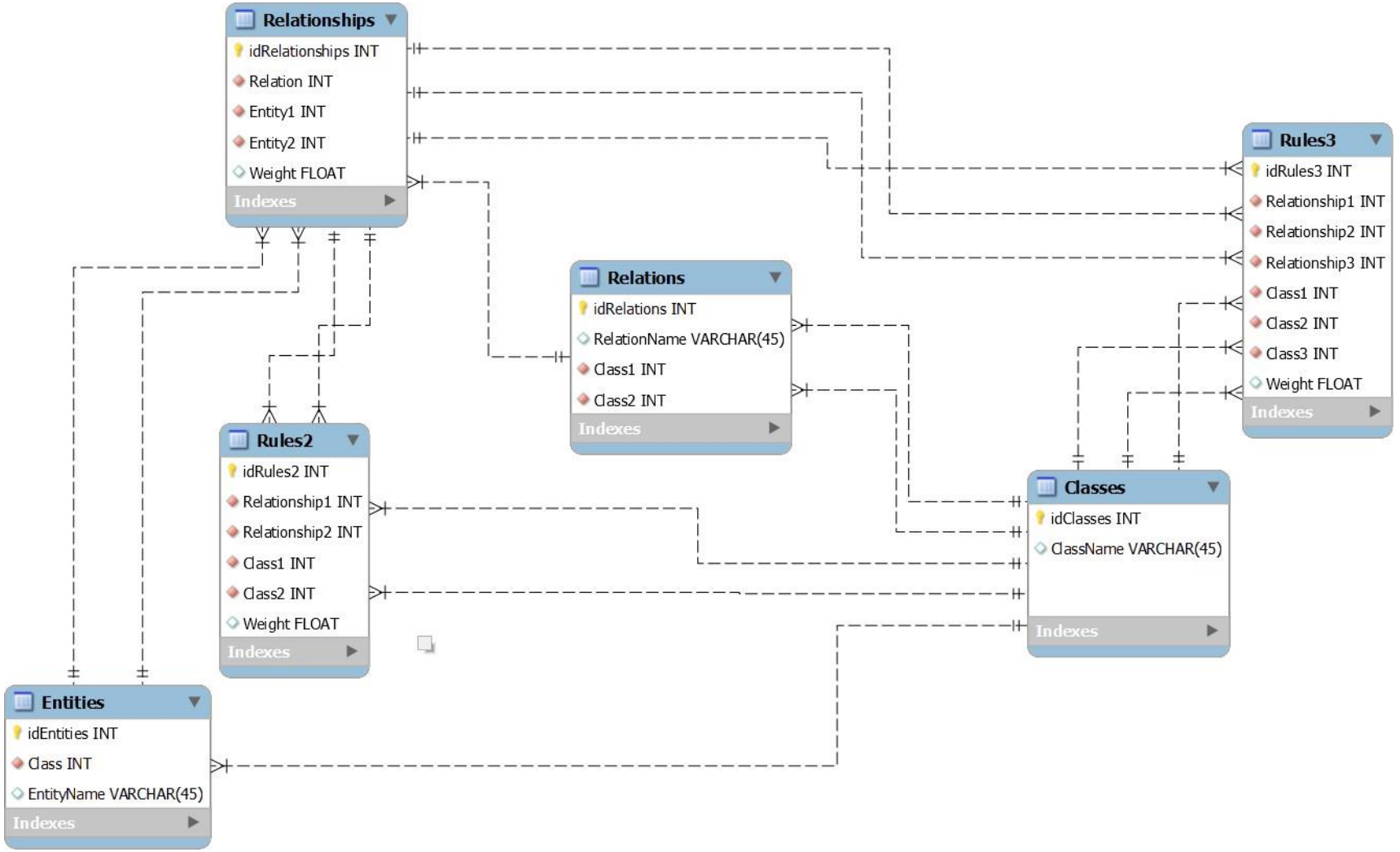

Рисунок 4.2 – Схема реляційної моделі бази знань.

«Місце». Таблиця Relations може відображати факти та можливий зв'язок між класами сутностей. Таблиця містить показники класів,які можуть бути використані для відображення цього зв'язку.

Таблиця Relationships містить зважену інформацію про факти. У цій таблиці є вага, що відображає ймовірність правди факту, що записаний Наприклад:

$$0{,}95 \text{ народився в}(\text{Володимир Зеленський}, \text{Дніпропетровськ}) \quad (4.2)$$

Таблиця Rules містить інформацію про правила виведення. Таблиця містить атрибути: ідентифікатори фактів, ідентифікатори класів та вагу. Ідентифікатори класів та фактів відображають, які класи та факти беруть участь у правилі. Вага відображає ступінь коректності правила.

Наприклад:

$$1.40\ \forall x \in W\ \forall y \in P\ (\text{жив в}(x, y) \leftarrow \text{народився в}(x, y)) \quad (4.3)$$

Елемент Rules не є єдиною таблицею. Для кожного з 6 можливих типів правил є своя таблиця. Можливі типи правил представлені нижче:

$$\forall x \in C1, y \in C2\ (p(x, y) \leftarrow q(x, y)), \quad (4.1.4)$$

$$\forall x \in C1, y \in C2\ (p(x, y) \leftarrow q(y, x)), \quad (4.1.5)$$

$$\forall x \in C1, y \in C2, z \in C3\ (p(x, y) \leftarrow q(z, x), r(z, y)), \quad (4.1.6)$$

$$\forall x \in C1, y \in C2, z \in C3\ (p(x, y) \leftarrow q(x, z), r(z, y)), \quad (4.1.7)$$

$$\forall x \in C1, y \in C2, z \in C3\ (p(x, y) \leftarrow q(z, x), r(y, z)), \quad (4.1.8)$$

$$\forall x \in C1, y \in C2, z \in C3\ (p(x, y) \leftarrow q(x, z), r(y, z)). \quad (4.1.9)$$

Тож кортежі для кожного з 6 правил будуть відповідно:

$$M_{1\_2} = (R1, R2, C1, C2, w), \quad (4.4)$$

$$M_{3\_6} = (R1, R2, R3, C1, C2, C3, w). \quad (4.5)$$

На етапі формування факторного графу виконуються такі запити до бази даних:

Наступні два запита на рисунках повертають об'єднання таблиць за правилом та класом сутності правила..

```
SELECT Rules2.Relationship1 AS R,
Relationships.Entity1 AS x, Relationships.Class1 AS C1
Relationships.Entity2 AS y, Relationships.Class2 AS C2
FROM Rules2
JOIN Relationships ON Rules2.Relationship2 = Relationships.Relation
AND Rules2.Class1 = Relationships.Class1
AND Rules2.Class2 = Relationships.Class2
```

Рисунок 4.3 – Запит, що виводить факти з правил першого типу

```
SELECT Rules3.Relationship1 AS R,
Relationships2.Entity2 AS x, Relationships2.Class2 AS C1,
Relationships3.Entity2 AS y, Relationships3.Class2 AS C2
FROM Rules3
JOIN Relationships Relationships2
ON Rules3.Relationship2 = Relationship2.Relation AND
Rules3.Class3 = Relationships2.Class1
AND Rules3.Class1 = Relationships2.Class2
JOIN Relationships Relationships3
ON M3.Relationship3 = Relationships3.Relation
AND Rules3.Class3 = Relationships3.Class1
AND Rules3.Class2 = Relationships3.Class2
WHERE Relationships2.x = Relationships3.x
```

Рисунок 4.4 – Запит, що виводить факти з правил третього типу

Запит на рисунку 4.5 повертає матрицю факторів. Він виконуєпоєднання таблиць правил та фактів будуючи матрицю зв'язків між правиламита фактами, що вони об'єднують.

```
SELECT Relationships1.idRelationships AS I1,
Relationships2.idRelationships AS I2,
Relationships3.idRelationships AS I3,
Rules3.w AS w
FROM Rules3
JOIN Relationships Relationships1 ON
Rules3.Relationship1 = Relationships1.Relation
AND Rules3.Class1 = Relationships1.Class1
AND Rules3.Class2 = Relationships1.Class2
JOIN Relationships Relationships2 ON
Rules3.Relationship2 = Relationships2.Relation
AND Rules3.Class3 = Relationships2.Class1
AND Rules3.Class1 = Relationships2.Class2
JOIN Relationships Relationships3
ON Rules3.Relationship3 = Relationships3.Relation
AND Rules3.Class3 = Relationships3.Class1
AND Rules3.Class2 = Relationships3.Class2
WHERE Relationships1.x = Relationships2.y
AND Relationships1.y = Relationships3.y
AND Relationships2.x = Relationships3.x;
```

Рисунок 4.5 – Запит, що будує матрицю факторного графу

Схема алгоритму роботи програми представлена на рисунку 4.6.

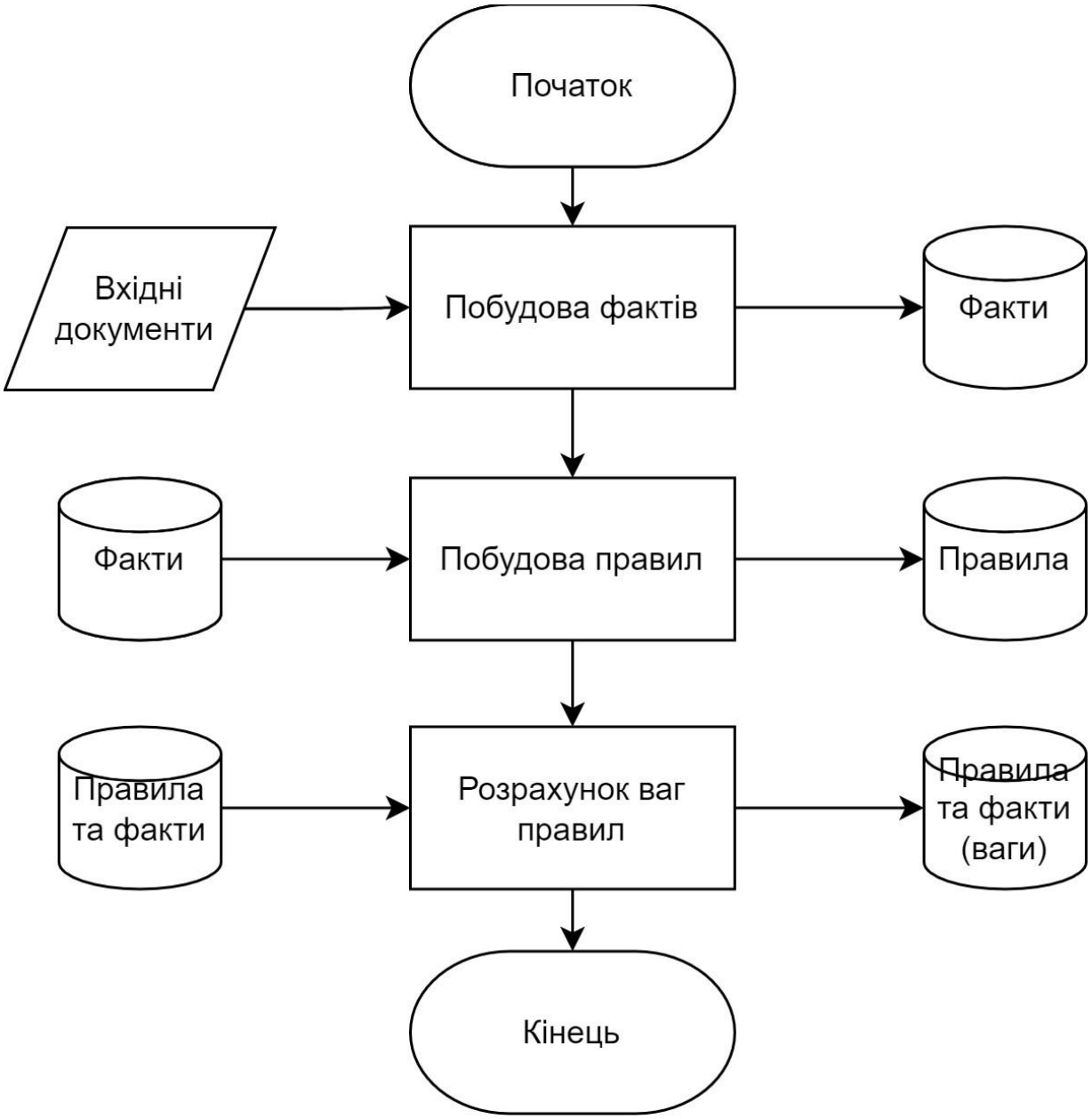

Рисунок 4.6 – Схема загального алгоритму удосконаленого методу

На рисунку 4.7 представлена екранна форма на етапі вилучення знань. Вона відображає поточні результати вилучення знань з джерел. На ній можна ініціювати додавання джерела та їх аналізу. На рисунку 4.8 представлена екранна форма етапу визначення типів відношень. На ній можна відсортувати правила та обрати вибрати необхідні. На цій формі можна ініціювати редагування обраних правил.

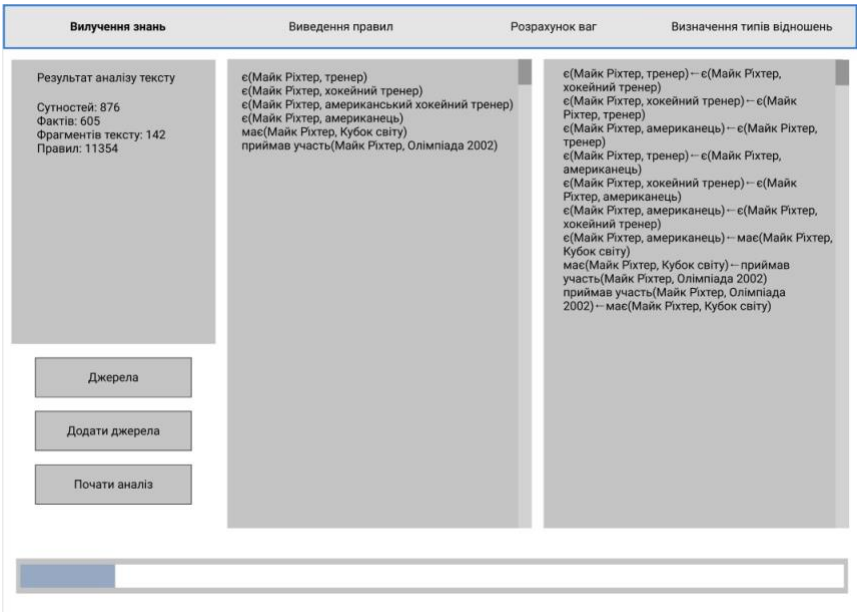

Рисунок 4.7 – Екранна форма на етапі вилучення знань

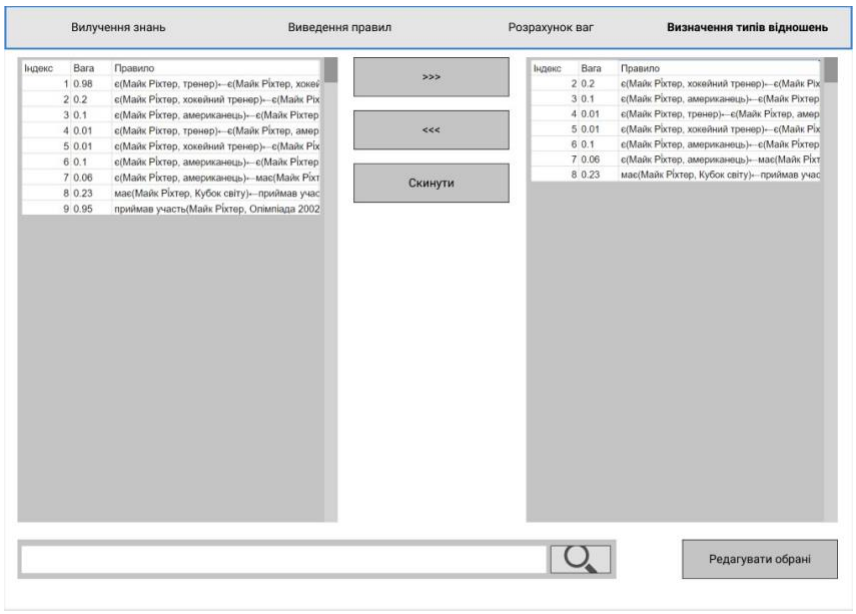

Рисунок 4.8 – Екранна форма на етапі визначення типів відношень

Таблиця 4.4 – Модель представлення знань

Сутності	Класи	Зв’язки	Взаємозв’язки
Володимир Зеленський	W (Політик) = {Володимир Зеленський}	народився в(W, P), народився в(W, C)	народився в(Володимир Зеленський, Дніпропетровськ)
Нью-Йорк	C (Місто) = {Дніпропетровськ }	жив в(W, P), жив в(W, C)	народився в(Володимир Зеленський, Кривий Ріг)
Бруклін	P (Місце) = {Кривий Ріг}	знаходиться в(P, C)	
Правила			
∀x ∈ W ∀y ∈ P (жив в(x, y) ← народився в(x, y))			
∀x ∈ W ∀y ∈ C (жив в(x, y) ← народився в(x, y))			
∀x ∈ P ∀y ∈ C ∀z ∈ W (знаходиться в(x, y) ← жив в(z, x) A жив в(z, y))			
∀x ∈ P ∀y ∈ C ∀z ∈ W (знаходиться в(x, y) ← народився в(z, x) A народився в(z, y))			
∞ ∀x ∈ C ∀y ∈ C ∀z ∈ W (народився в(z, x) A народився в(z, y) → x = y)			

4.2 Експериментальна перевірка методу

Для порівнянні базового та удосконаленого методів ми використовували показники: кількість вилучених фактів, кількість вилучених правил. На рисунку 4.9 представлена діаграма порівняння у системі ProbKB та при використані удосконаленого методу.

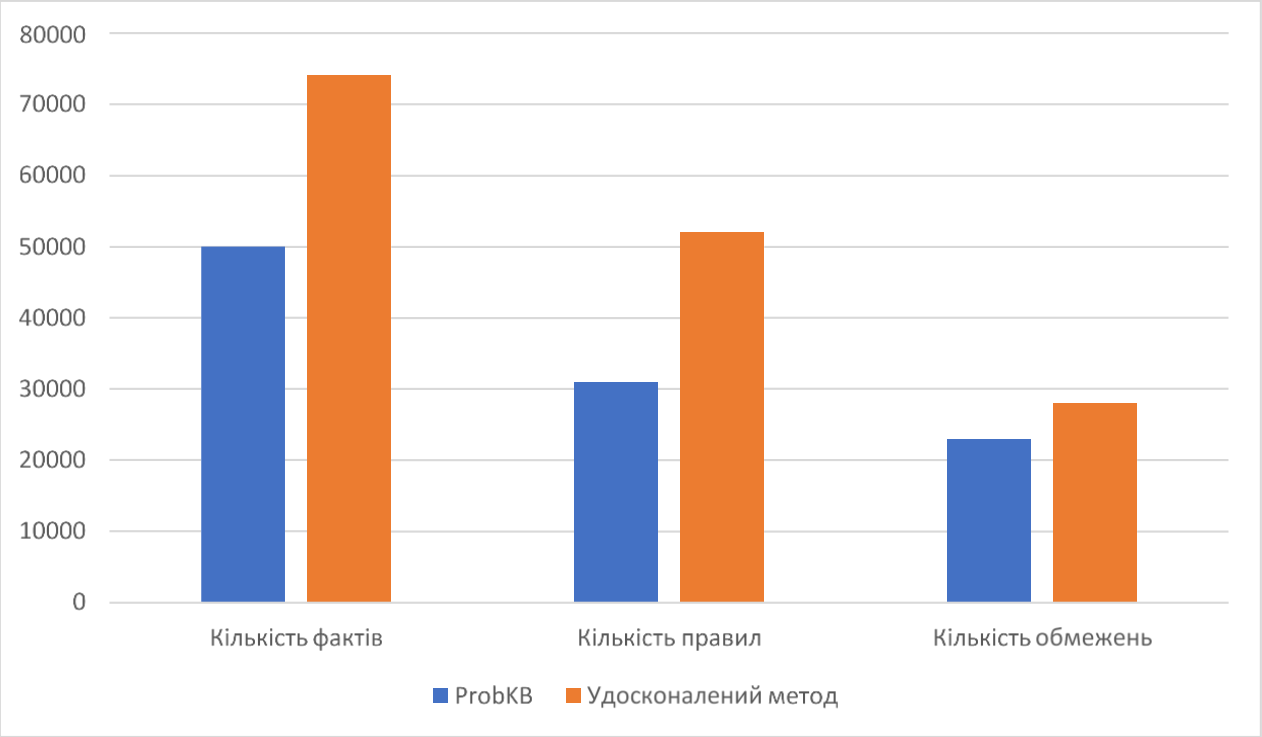

Рисунок 4.9 – Діаграма порівняння методів ProbKB та удосконаленого методу автоматизованої побудови БЗ

Діаграма відображає, що при використанні удосконаленого методу використовувалась більша кількість правил та обмежень для виведення, ніж кількість правил у сеті, що використовує система ProbKB. Також, завдяки більшій кількості правил та обмежень спостерігається більша кількість фактів,що були виведені під час роботи удосконаленого методу.

Отже, удосконалений метод показав кращі результати ніж ProbKB. Удосконалений метод виводить правила та обмеження для виведення фактів із текстів, що аналізуються. Це дозволяє збирати усі правила, що можуть бути застосовані до цих фактів, під час аналізу тексту, що в свою чергу, збільшує кількість фактів, що можуть бути виведені в майбутньому.

ВИСНОВКИ

Сучасні бази знань мають і систематизують знання з багатьох різних тематичних областей. Окрім інформації про певну сутність, вони також зберігають інформацію про їхні стосунки між собою.

Оскільки публічні бази знань часто не містять необхідної інформації для аналізу таких сценаріїв виникає потреба у створенні та підтримці спеціалізованих баз знань для конкретної області.

У цій дипломній роботі досліджується метод автоматизованої побудови бази знань на основі аналізу її поведінки, що представлена у вигляді записів про послідовність виконання дій бізнес-процесів.

При автоматизованому управлінні базами знань здійснюється спільний процес управління знаннями, якому необхідний розбір і аналіз моделей процесу .

Процес управління знаннями утворюється з етапів визначення практичної необхідності у знаннях, визначення джерел отримання знань, вилучення та збереження знань, а також застосування знань. Втілення цих етапів при вирішенні задач підтримки управлінських рішень значно залежить від характеристик знань, що використовуються в організації. Відповідним чином, методи автоматизованого управління БЗ враховують специфіку організаційних знань, та основуються на моделях процесу управління цими знаннями.

Було проаналізовано метод автоматизованої побудови бази знань на основі факторного графу. Метод полягає у тому, щоб побудувати факторний граф для імовірнісного розподілення використовуючи SQL запити. Для розрахунку розподілення використовувалося розподілення Гіббса. Розглянутий метод використовує статичний сет правил. Це обмежує можливості висновку інформаційно-довідкової системи на основі цього методу.

Розроблений метод поєднує вирішення задач поповнення бази знань, виявлення аномального стану об'єкту управління, а також формування багатоваріантного управлінського рішення. Метод орієнтований на інтерактивну реалізацію при виникненні нових станів об'єкту управління, що дає можливість будувати нові послідовності управляючих дій з використанням актуального набору зважених темпоральних правил.

Для описаної технології розроблено програмне забезпечення мовою C#. З використанням розробленого програмного забезпечення був проведений експеримент та виконаний порівняльний аналіз системи з використанням удосконаленого методу та методу на основі факторного графу. По результатам експерименту було виявлено, що удосконалений метод може виявляти більше правил, ніж використовується в базовому, а також може виводити більше фактів.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Методичні вказівки щодо розробки та оформлення кваліфікаційної роботи (для студентів усіх форм навчання другого (магістерського) рівня вищої освіти спеціальності 122 Комп'ютерні науки освітньо-професійної програми «Інформаційні управляючі системи та технології») / Упоряд.: Петров К.Е., Левикін В.М., Чалий С.Ф., Євланов М.В., Саєнко В.І., Міхнов Д.К., Міхнова А.В., Чала О.В. – Харків: ХНУРЕ, 2021. – 30 с.

2. ДСТУ 3008-95 Документація. Звіти у сфері науки і техніки. Структура і правила оформлення / Державний стандарт України. Київ : Державний комітет України по стандартизації, метрології та сертифікації, 1996. 29с.

3. https://medium.com/@CereLabs/helplessness-of-software-d83c984ac6a7 [Електронний ресурс]. – 2022. – Режим доступу до ресурсу: https://medium.com/@CereLabs/helplessness-of-software-d83c984ac6a7.

4. YAGO: A Multilingual Knowledge Base from Wikipedia, Wordnet, and Geonames / T.Rebele, F. Suchanek, J. Hoffart, J. Biega. // Lecture Notes in Computer Science. – 2016.

5. Brachman R. J. KNOWLEDGE REPRESENTATION AND REASONING / R. J. Brachman, H. J. Levesque. – San Francisco: Elsevier, 2004. –413 с.

6. https://en.wikipedia.org/wiki/Never-Ending_Language_Learning [Електронний ресурс] – Режим доступу до ресурсу: Never-Ending LanguageLearning.

7. Ratner A. DeepDive: Declarative Knowledge Base Construction [Електронний ресурс] / A. Ratner, J. Shin, F. Wang. – 2015. – Режим доступу до ресурсу:

tps://sigmodrecord.org/publications/sigmodRecord/1603/pdfs/16_DeepDive_RH

8. The Dark Data Rises [Електронний ресурс] – Режим доступу до ресурсу: https://medium.com/@CereLabs/the-dark-data-rises-f911422d80fc.

9. Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction – 209 N. Eighth Street, 2012. –139 с.

10. Пасічник В. В. Організація баз даних та знань / В. В. Пасічник, В.А. Резніченко. – Київ, 2006. – 378 с.

11. Chen Y. Knowledge expansion over probabilistic knowledge bases / Y.Chen, D. Zhe Wang. // SIGMOD '14. – 2014.

12. Elementary: Large-scale Knowledge-base Construction via Machine Learning and Statistical Inference. // Semantic Web and Information Systems – Special Issue on Web-Scale Knowledge Extraction. – 2012. – С. 23.

13. Milton N. Knowledge Acquisition in Practice: A Step-by-step Guide / Nicholas Milton. // Knowledge Acquisition in Practice. – 2007. – С. 43.

14. Ji S. A Survey on Knowledge Graphs: Representation, Acquisition and Applications / S. Ji, E. Cambria, P. Marttinen. // JOURNAL OF LATEX CLASS FILES. – 2015. – №8. – С. 27.

15. Rudin F. Falling Rule Lists / F. Rudin, C. Wang. // JOURNAL OFLATEX CLASS FILES. – С. 10.

16. Martinez-Gil J. Automated knowledge base management: A survey / Jorge Martinez-Gil. // Computer Science Review. – 2015.

17. Suchanek F. YAGO: A Core of Semantic Knowledge Unifying WordNet and Wikipedia / F. Suchanek, G. Kasneci, G. Weikum. // Track: SemanticWeb. – 2007.

18. R. W. Blanning, S. Ram, and R. Y. Wang, «Information technologies and systems», Decision support systems, vol. 13, no. 3–4, pp. 219–221, 1995, doi: 10.1016/0167-9236(93)E0043-D.

19. H. Mintzberg, D. Raisinghani, and A. Theoret, «The structure of «Unstructured» decision processes», Administrative science quarterly, vol.

21, no. 2, pp. 246-248, 1976, doi: 10.2307/2392045.

20. Т.В. Козуля, Н.В. Шаронова, М.М. Козуля, « Формування знанняорієнтованого інформаційного забезпечення досліджень складних 327 систем», Системні дослідження та інформаційні технології, №. 7, с. 63– 72, 2017, https://doi.org/10.20535/SRIT.2308-8893.2017.3.07.

21. H. Uehara, T. Yamaguchi, and Q. Bai, Knowledge management and acquisition for intelligent systems in 17th Pacific rim knowledge acquisition workshop, PKAW 2020, Yokohama, Japan, 2021. Cham: Springe

22. T. H. Davenport и L. Prusak, Working knowledge: How organizations manage what they know. Boston, MA: Harvard business school press, 2000.

23. F. M. Suchanek, J. Lajus, A. Boschin, and G. Weikum, «Knowledge representation and rule mining in entity-centric knowledge bases», in Reasoning Web. Explainable Artificial Intelligence, no. 11810, M. Krötzsch 330 and D. Stepanova, Eds. Cham: Springer International Publishing, 2019, pp. 110–152. doi: 10.1007/978-3-030-31423-1_4.

24. R.K. Bali, N. Wickramasinghe, and B. Lehaney, Knowledge management primer. New York: Routledge, 2009.

25. P. R. Gamble and J. Blackwell, Knowledge management: A state of theart guide. Kogan Page Ltd, 2001

26. S. Wang and D. Liu, «Knowledge representation and reasoning for qualitative spatial change», Knowledge-Based Systems, vol. 30, pp. 161–171 2012,doi: 10.1016/j.knosys.2012.01.009

27. I. Hatzilygeroudis and J. Prentzas, «Knowledge representation requirements for intelligent tutoring systems», in ITS 2004. International Conference on Intelligent Tutoring Systems. Berlin, Heidelberg, 2004, vol. 3220, pp. 87–97.

28. Q. Wang, Z. Mao, B. Wang, and L. Guo, «Knowledge Graph Embedding: A Survey of Approaches and Applications», in IEEE Transactions on Knowledge and Data Engineering, vol. 29, no. 12, pp. 2724–2743, 2017, doi:

10.1109/TKDE.2017.2754499.

29. A. Felfernig and F. Wotawa, «Intelligent engineering techniques for knowledge bases», Ai Communications, vol. 26, no. 1, pp. 1–2, 2013, doi: 10.3233/AIC-2012-0541.

30. Левикін В. М., Чала О.В. Модель бази знань інформаційної системи процесного управління. Вісник Національного технічного університету «ХПІ». Системний аналіз, управління та інформаційні технології. 2017. № 28(1250). С. 74-78.

31. Чала О.В. Принцип та метод еволюційної побудови бази знань на основі аналізу логів ІС процесного управління. Біоніка інтелекту. 2017. № 1(88). С. 80-84.

32. Левикін В. М., Чала О. Концепція автоматизованої побудови бази знань у системі процесного управління. Біоніка інтелекту. 2017. № 2(89). С. 77-83.

33. Левикін В. М., Чала О. В. Розробка представлення причинно-наслідкових залежностей для бази знань системи процесного управління. Вісник Національного технічного університету «ХПІ»: Системний аналіз, управління та інформаційні технології. 2018. № 21(1297). С. 48-53.

34. Чала О. В. Принципи автоматизованої побудови та використання темпоральної бази знань. Системи управління, навігації та зв’язку. 2018. № 6 (52). С. 122-125. DOI: 10.26906/SUNZ.2018.6.122.

35. Чала О.В. Побудова подієвої складової бази знань в рамках системи управління підприємством. Інформаційні технології – 2018 (ІТ-2018): Тези допов. V Всеукр. наук.-практ. конф. (Київ, 17 трав. 2018). Київ: УГ. С. 143-145.

36. Левикін В.М., Чала О.В. Контекстні обмеження в базі знань інформаційної системи процесного управління. Комп’ютерні та інформаційні системи і технології: Тези допов. ІІ Міжнар. наук.-техн. конф. (Харків, 18-19 квіт. 2018). Харків, ХНУРЕ, 2018. С. 108.