МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет кораблебудування імені адмірала Макарова

Навчально-науковий інститут комп'ютерних наук та управління проектами
Кафедра інформаційних управляючих систем та технологій

"Допущений до захисту"

Завідувач кафедри ІУСТ

_____

к.т.н, доц. Михелєв І.Л.

"____" _____ 2025 р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття ступеня вищої освіти "бакалавр"

на тему: **Побудова пошукової WEB – системи для предметно – орієнтованих експертних систем**

Виконав: студент групи

Андрєєв Д.А.

Керівник роботи:

Професор каф. комп'ютерних технологій та інформаційної безпеки д.т.н., доц.,

Передерій В.І.

м. Миколаїв – 2025 р.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет кораблебудування імені адмірала Макарова

Навчально-науковий інститут комп'ютерних наук та управління проектами

Кафедра інформаційних управляючих систем та технологій

Спеціальність 122 "Комп'ютерні науки"

Освітня програма "Комп'ютерні науки"

"ЗАТВЕРДЖУЮ"

Гарант освітньої програми

_____

к.т.н, доц. Гайда А.Ю.

"____" _____ 2025 р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття ступеня вищої освіти "бакалавр"

Студенту Андрєєву Дмитру Андрійовичу

1. Тема роботи: Побудова пошукової WEB – системи для предметно – орієнтованих експертних систем

Керівник роботи д.т.н., доц. Передерій В.І.

Затверджені наказом ректора № 295-уч від "27" березня 2025 року.

2. Термін подання роботи: 20 червня 2025 р.

3. Вихідні дані до проекту (роботи) ДСТУ щодо обробки інформації, літературні джерела, технічна документація на існуючі аналоги систем,

4. Перелік питань, що належать до розробки (найменування розділів)

1. Визначити основні особливості пошукових web - систем в предметно – орієнтованих експертних системах.

2. Провести аналіз сучасних пошукових систем та їх характеристики.

3.Розробити математичну модель процесу пошуку та оцінки релевантних документів з WEB середовища.

4. Розробити програмно – алгоритмічного забезпечення пошукової WEB – системи.

5. Перелік презентаційних матеріалів_наведено в слайдах

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1	Передерій В.І	03.02.2025	22.02.2025
2	Передерій В.І	22.02.2025	21.03.2025
3	Передерій В.І	21.03.2025	25.04.2025
4	Передерій В.І	25.04.2025	07.06.2025

7. Дата видачі завдання 03 лютого 2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів роботи	Примітка
1	Аналіз предметної галузі	13.02.2025	Виконав
2	Постановка задачі	15.02.2025	Виконав
3	Розробка концепції	22.02.2025	Виконав
4	Розробка проєкту ІС	21.03.2025	Виконав
5	Реалізація проєкту	25.04.2025	Виконав
6	Розробка питань з охорони праці	07.06.2025	Виконав
7	Розробка графічних матеріалів	12.06.2025	Виконав
8	Подання роботи на перевірку на плагіат	10.06.2025	Виконав
9	Подання роботи рецензенту	15.06.2025	Виконав
10	Подання роботи на захист	20.06.2025	Виконав

Студент Андрєєв Д.А.

Керівник роботи Передерій В.І

Анотація

У роботі представлено побудову пошукової WEB – системи для предметно – орієнтованих експертних систем.

Основною метою роботи є розробка системи пошуку та оцінки релевантних документів при побудові предметно–орієнтованої експертної системи з використанням інтернет технологій. Для рішення даного завдання були визначені основні фактори впливу на процес пошуку інформації та проведена оцінка її релевантності.

У роботі розроблені математичні моделі процесу пошуку з застосуванням теорії експертних оцінок та ймовірнісного методу Байесовської мережі довіри, як міри релевантності документу запиту користувача.

Для практичної реалізації системи пошуку розроблені алгоритми роботи та структура адаптивної бази даних де згенерована інформаційна потреба з відповідними критеріями пошуку в Інтернеті.

Побудована пошукова WEB – система може бути використана в управлінських, бізнесових, наукових сферах та інших предметно – орієнтованих експертних систем для підвищення якості прийняття рішень.

Ключові слова: пошукові WEB – системи, предметно – орієнтовані експертні системи, система підтримки прийняття рішень, Баєсовська мережа довіри, теорія експертних оцінок, релевантність документу, програмне середовище.

Abstract

The work presents the construction of a search WEB system for subject-oriented expert systems.

The main goal of the work is to develop a system for searching and evaluating relevant documents when building a subject-oriented expert system using Internet technologies. To solve this problem, the main factors influencing the information search process were identified and its relevance was assessed.

The work developed mathematical models of the search process using the theory of expert assessments and the probabilistic method of the Bayesian trust network as a measure of the relevance of the user's request document.

For the practical implementation of the search system, algorithms and the structure of an adaptive database were developed where the information need with the appropriate search criteria on the Internet is generated.

The constructed search WEB system can be used in management, business, scientific areas and other subject-oriented expert systems to improve the quality of decision-making.

Keywords: search WEB systems, subject-oriented expert systems, decision support system, Bayesian trust network, expert evaluation theory, document relevance, software environment.

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

БМД - Баєсовська мережа довіри.

СППР - Система підтримки прийняття рішень

ШІ (AI) - Штучний інтелект

ПАВС - Пошукові адаптивні веб-системи

АПВС - Адаптивні пошукові веб-системи

JS – Java Script

ПС – Програмне середовище

ЗМІСТ

ВСТУП

На сьогодні одним із видів інтелектуальних технологій є експертні системи, що орієнтуються на здобуття, обробку і використання додаткової інформації - знань. Ці інтелектуальні програми, здатні здійснювати логічні виведення на підставі знань у конкретній предметній області та забезпечувати рішення специфічних задач на професійному рівні. Технології штучного інтелекту сприяли створенню саме таких систем. Необхідність їх створення була викликана недостатньою кількістю фахівців-експертів, які могли б у будь-який час кваліфіковано відповідати на питання своєї предметній області.

Хоча звичайно експертні системи спрямовані на використання у досить обмежених предметних галузях, у випадку застосування їх для вирішення реальних завдань досягнуті значні результати. Вони обумовили велике зацікавлення експертними системами не тільки фахівців-теоретиків, але й практичних працівників у найрізноманітніших галузях людської діяльності

Актуальність теми. На сьогодні пошукові WEB - системи відіграють ключову роль у сучасних предметно – орієнтованих експертних системах, забезпечуючи ефективний пошук, аналіз та подання інформації для прийняття обґрунтованих рішень. Вони є невід’ємним елементом управління великими обсягами даних, що сприяє автоматизації рутинних процесів, підвищенню точності аналітики та адаптації до індивідуальних потреб користувачів.

Зважаючи на це, у даній роботі пропонується розробка системи пошуку та оцінки релевантних документів при побудові предметно–орієнтованої експертної системи з використанням WEB технологій для підвищення якості прийняття рішень.

Мета дослідження. Метою роботи є розробка системи пошуку та оцінки релевантних документів при побудові предметно–орієнтованої експертної системи з використанням WEB - технологій.

Завдання дослідження:

- Провести аналіз сучасних підходів до побудови предметно–орієнтованих експертних систем.
- Визначити основні фактори впливу на процес пошуку інформації.
- Дослідити та провести оцінку основних факторів впливу по ступеню важливості використовуючи знання експертів-фахівців.
- Розробити математичну модель процесу пошуку інформації та провести оцінку міри релевантності документу запиту користувача з застосуванням ймовірнісного методу Байесовської мережі довіри.
- Розробити програмно-інструментальні засоби для реалізації системи пошуку та оцінки релевантних документів при побудові предметно–орієнтованої експертної системи з використанням WEB - технологій.

Об'єкт дослідження: пошукові предметно–орієнтовані експертні системи в складі СППР.

Предмет дослідження: моделі, методи та технології розробки пошукових предметно–орієнтованих експертних систем в складі СППР.

Методи дослідження: системний аналіз, теорія експертних систем, ймовірнісний метод Байесовська мережа довіри, інформаційні WEB -технології.

Наукова новизна одержаних результатів. Розроблена математична модель і алгоритми процесу пошуку та оцінки релевантних документів з WEB середовища для предметно–орієнтованих експертних систем в складі СППР.

Очікуваний результат. Результати побудови пошукової WEB – системи можуть бути використані в управлінських, бізнесових, наукових сферах та інших предметно – орієнтованих експертних системах для підвищення якості прийняття рішень.

Бакалаврська робота є самостійним дослідженням, яке включає теоретичні розділи, аналітичну частину, розробку практичних рекомендацій та оцінку їх ефективності.

Робота базується на актуальних наукових публікаціях і результатах експериментальних досліджень.

РОЗДІЛ 1. ОСОБЛИВОСТІ ПОШУКОВИХ WEB - СИСТЕМ В ПРЕДМЕТНО – ОРІЄНТОВАНИХ ЕКСПЕРТНИХ СИСТЕМАХ

1.1 Особливості предметно – орієнтованих експертних систем.

На даний час експертні системи представляють собою широкий вибір програмно апаратних засобів, які класифікуються за різними ознаками.

Для розв'язання типових задач, орієнтованих на конкретну предметну область, і які важко розв'язувати за допомогою традиційних математичних методів, залучаються методи та засоби експертних систем представлені на рис. 1.1. До рішення цих задач відносяться:

– витягнення узгодженої інформації з первинних даних;

– виявлення несправностей і причин їх появи в деякій системі;

– безперервна інтерпретація даних в режимі реального часу із сигналізацією виходу контрольованих параметрів за допустимі межі;

– передбачення ймовірних наслідків на основі минулих і поточних подій;

– визначення послідовності дій, направлених на досягнення попередньо поставлених цілей;

– визначення конфігурації технічної системи за заданих обмежень;

– визначення послідовності дій з приведення системи до необхідних режимів функціонування;

– інтерпретація, діагностика і корекція знань та умінь учнів;

– формування керуючих впливів, які визначають процеси функціонування складних технічних систем;

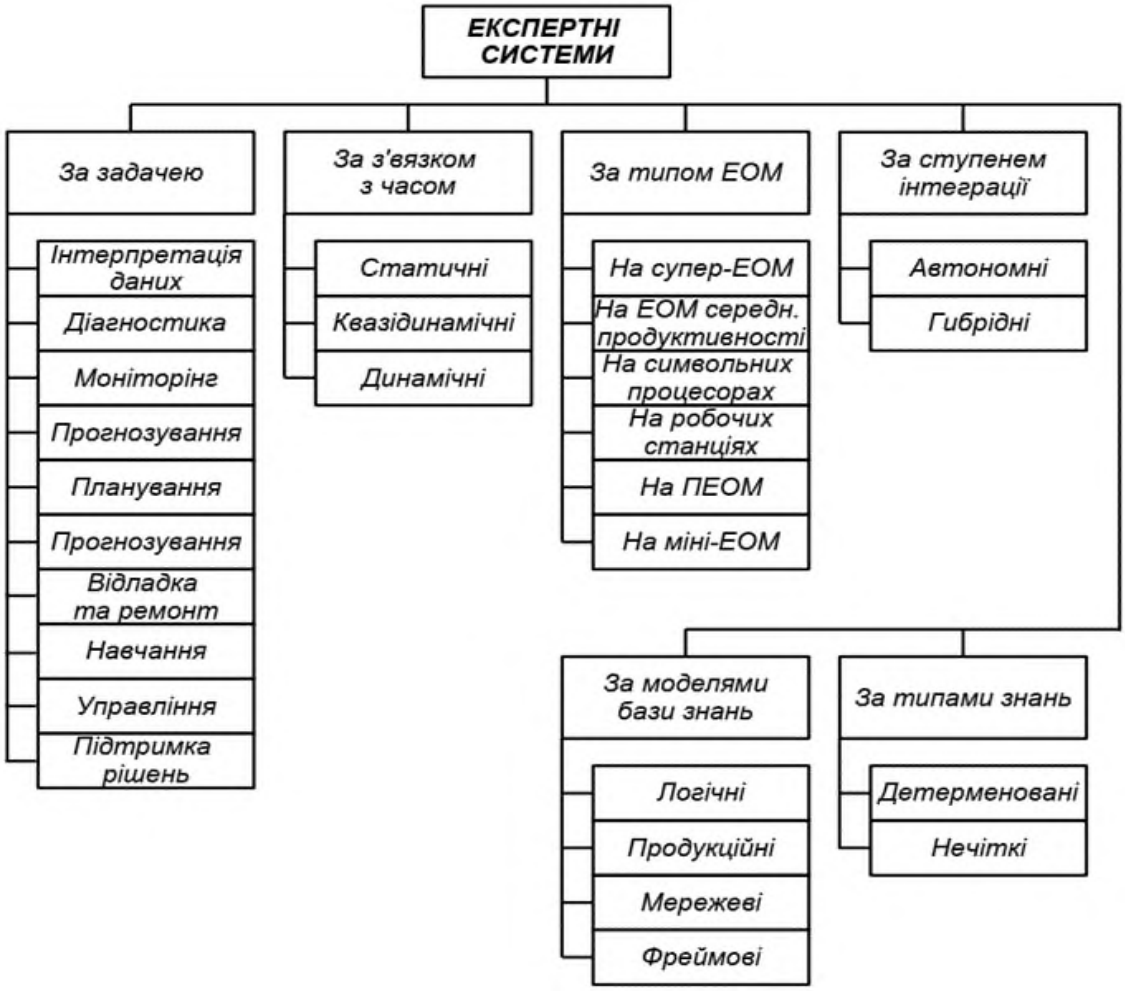

Рисунок 1.1 - Класифікація експертних систем розв'язання типових задач.

– забезпечення особи, яка приймає рішення, необхідною інформацією та рекомендаціями для полегшення процесу прийняття рішень.

За ознаками експертні системи класифікація на:

- статичні експертні системи передбачають незмінність часу, статичність моделі предметної області;

- динамічні експертні системи функціюють у сполученні із датчиками об'єктів у режимі реального часу з безперервною інтерпретацією даних, які поступають в систему;

Основні задачі предметно-орієнтованих експертних систем […]:

1. Можливість акумуляції експертних знань, отриманих з декількох джерел., що дає можливість зібрати та використовувати знання багатьох експертів у предметній області.

2. Зберігання в базі даних експертних знань висококваліфікованих експертів-фахівців.

3. Не залежність ефективної роботи експертної системи від психо-функціонального та психо-емоційного стану людини-експерта, що виключає вплив когнітивних факторів на процес прийняття рішення.

4. Доступність інформації до експертних знань з використанням будь-яких наявних комп'ютерних засобів.

5. Зниження витрат на отримання експертних знань.

6. Можливість пояснення та отримання результату, під час вирішення задачі експертних знань.

7. Можливість використання інтелектуальних навчальних програм шляхом діагностики та навчання в предметній області.

8. Використовувати доступ до бази даних з використанням алгоритмів штучного та гібридного інтелекту.

На сьогодні експертні системи знаходять широке застосування в багатьох областях: в енергетиці, медицині, управлінні, науці та інш. Експертні системи стали незамінними внаслідок їх здатності накопичувати знання та адаптуватись для сприйняття інформації в конкретній предметній області.

1.2 Особливості пошукових задач, та засоби і технології інформаційного пошуку в предметно – орієнтованих експертних системах.

Як відомо пошук інформації це процес виявлення релевантних документів у відповідній предметно–орієнтованій темі.

Рисунок 1.2 – Структурна схема інформаційного пошуку в предметно – орієнтованих експертних системах.

Основною задачею інформаційного пошуку являється допомога оператору задовольнити інформаційну потребу, які формулюються як запит, що представляє собою набір ключових слів.

Наведемо список завдань інформаційного пошуку що включає […]:

– моделювання;

– класифікацію пошукових документів-запитів;

– фільтрацію пошукових документів-запитів;

– кластеризація документів-запитів;

– проектування пошукових систем і адаптивних інтерфейсів;

– отримання документів-запитів;

– мови запитів та ін.

Процес пошуку документів-запитів складається з наступних етапів […]:

– формулювання запиту, вибір пошукових системи і сервісів;

– пошук в декількох пошукових системах;

– отримання результатів документів-запитів;

– обробка отриманих результатів та їх корекція;

– модифікація документів-запитів з подальшою обробкою отриманих результатів.

Методами пошуку документів-запитів являються задачі відбору документів, що відображають процес знаходження рішення та залежать від проблем предметної області.

Процес пошуку документів-запитів розглядається як іттеративний процес, що утворює методологічну основу стратегії пошуку яку можна розділити на наступні класи […]:

– в заданому предметно-орієнтованому просторі;

– в ієрархічно тематичному просторі;

–в альтернативних тематичних просторах;

– в динамічному процесі пошуку.

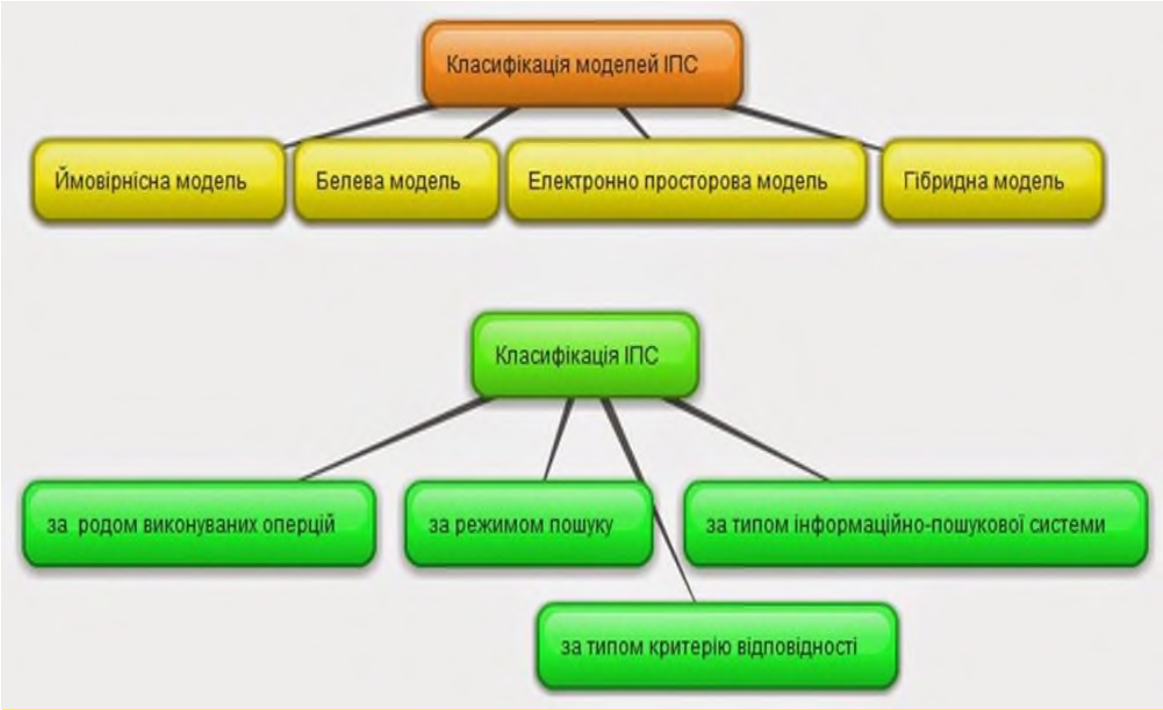

Рисунок 1.3 - Класифікація моделей інформаційного пошуку документів-запитів в предметно – орієнтованих експертних системах.

Розглянемо наступні види пошуку:

1. Повнотекстовий пошук – це пошук по всьому змісту документа-запиту, що використовує попередньо побудовані індекси якими є інвертовані індекси.

2. Пошук документів-запитів по метаданих – це пошук що підтримується системою – назва документа, дата створення, розмір, автор тощо.

3. Пошук по зображенню – це пошук документів-запитів за змістом зображення. Прикладами таких пошукових систем є Xcavator, Retrievr, PolarRose, Picollator Onlineby Recogmission. Google.

3. Компоненти інформаційного пошуку документів-запитів.

В даному випадку процедури інформаційного пошуку документів-запитів виокремлюються на – абстрактний і конкретний. Абстрактний це сукупність інформаційного пошуку документів-запитів, правил індексування та критеріїв видачі до відповідності інформації.

Враховуючи особливості зберігання інформації процедура інформаційного пошуку документів-запитів розділяється на:

– семантичне осмислення документів-запитів, та видача відповідних адрес;

– пошук знаходження самих документів-запитів.

Однією з ключових характеристик пошукових веб-систем є їх здатність динамічно підлаштовуватися під вимоги конкретного користувача. Це досягається завдяки впровадженню технологій машинного навчання та обробки природної мови, які дозволяють:

- аналізувати історію пошукових запитів, поведінкові патерни та переваги користувачів;
- пропонувати персоналізовані результати, релевантні запитам;
- підтримувати багатомовність та враховувати культурні особливості.

Рисунок 1.4 - Технології інформаційного пошуку в предметно – орієнтованих експертних системах.

Завдяки адаптивності такі системи ефективно вирішують задачі як у корпоративному середовищі, так і в індивідуальному використанні.

Такі системи орієнтовані на роботу з великими обсягами даних, які можуть бути структурованими (таблиці, бази даних) та неструктурованими (тексти, зображення, відео). Основні особливості інтеграції з Big Data включають:

- автоматичний збір, агрегацію та очищення даних із різних джерел;
- використання кластерних обчислень для швидкої обробки великих масивів інформації;
- виявлення закономірностей і трендів для підтримки прогнозування.

Що дозволяє організаціям приймати рішення на основі аналітики реальних даних.

Однією з унікальних властивостей адаптивних пошукових систем є підтримка роботи з таким широким спектром даних як текстова інформація (наукові статті, звіти, документи), візуальні дані (зображення, відео, графіки), дані, отримані від сенсорів та пристроїв.

Це забезпечує універсальність систем, дозволяючи використовувати їх у таких галузях, як медицина (для аналізу медичних зображень), освіта (аналіз навчальних ресурсів) чи фінанси (обробка економічних показників).

Для забезпечення максимальної точності адаптивні пошукові системи використовують алгоритми ранжування результатів, які враховують:

- контекст запиту;
- авторитетність джерела;
- актуальність даних.

Системи застосовують методи фільтрації шуму, відсіюючи нерелевантну або застарілу інформацію. Таким чином, користувач отримує тільки ті дані, які дійсно впливають на процес прийняття рішення.

Сучасні адаптивні пошукові систем інтегрують інструменти для візуалізації даних, що дозволяє представити складну інформацію у вигляді графіків, діаграм або карт що сприяє швидшому розумінню великих масивів даних, ефективному порівнянню альтернативних сценаріїв та підготовці презентаційних матеріалів для командного обговорення.

Адаптивні пошукові системи дозволяють моделювати можливі сценарії розвитку подій. За допомогою аналізу "що-якщо" (What-If Analysis) користувачі можуть:

- оцінювати ризики впровадження нових стратегій;
- порівнювати кілька варіантів рішень;
- прогнозувати результати на основі історичних даних.

Це особливо актуально для галузей, де ризик є суттєвим фактором, наприклад, у фінансах чи виробництві.

Застосування алгоритмів штучного інтелекту забезпечує автоматизацію процесів аналізу:

- системи самостійно агрегують дані, аналізують їх та створюють аналітичні звіти;
- уникається необхідність ручної обробки, що скорочує час ухвалення рішень;

- автоматизація забезпечує високу точність і зменшує ймовірність людських помилок.

Пошукові адаптивні веб-системи базуються на хмарних технологіях, що забезпечують доступність із будь-якої точки світу, інтеграцію з корпоративними інформаційними системами (CRM, ERP, BI) та мають масштабованість, яка дозволяє адаптувати систему до зростаючих потреб організації.

Забезпечення безпеки даних є пріоритетом для сучасних адаптивних пошукових систем, особливо при роботі з конфіденційною інформацією. Системи використовують:

- протоколи шифрування;
- багатофакторну аутентифікацію;
- обмеження доступу до чутливих даних.

Це важливо для сфер, де конфіденційність є критичною, таких як медицина чи юриспруденція.

Основні особливості пошукових адаптивних веб-систем демонструють їхній значний вплив на функціональність систем підтримки прийняття рішень. Їх застосування дозволяє оптимізувати процеси обробки інформації, підвищувати точність прогнозів та приймати обґрунтовані рішення. Впровадження таких систем є необхідним для організацій, які прагнуть залишатися конкурентоспроможними у динамічному інформаційному

1.3. Роль та значення пошукових адаптивних WEB - систем в СППР.

Сучасні пошукові адаптивні веб-системи є важливим компонентом систем підтримки прийняття рішень (СППР). Вони спрямовані на збір, аналіз, фільтрацію та інтерпретацію даних з різних джерел для створення рекомендацій або прийняття ефективних рішень. Завдяки адаптивності такі системи здатні підлаштовуватися під потреби користувача, забезпечуючи персоналізовану взаємодію та підвищуючи ефективність аналізу.

Роль пошукових адаптивних веб-систем у СППР:

1. Автоматизація пошуку даних:

- Пошукові адаптивні системи дозволяють швидко знаходити потрібну інформацію з великої кількості джерел (наукові статті, бази даних, корпоративні сховища тощо).

- Вони автоматизують процес збору даних, зменшуючи витрати часу на ручний пошук.

2. Обробка неструктурованих даних:

- Системи підтримують роботу з різними форматами даних: текстами, зображеннями, відео, таблицями тощо.

- Вони можуть структурувати дані для подальшого аналізу, використовуючи методи машинного навчання (ML) та обробки природної мови (NLP).

3. Персоналізація:

- Адаптивні пошукові системи враховують історію запитів, переваги та поведінкові патерни користувачів.

- Це дозволяє створювати унікальний досвід для кожного користувача, що значно підвищує ефективність роботи.

4. Підтримка сценарного аналізу:

- Вони надають інструменти для прогнозування можливих результатів різних сценаріїв, використовуючи великі набори історичних даних.

- Наприклад, бізнес-аналітики можуть оцінювати ефективність стратегій продажів або вплив рішень на фінансові показники.

5. Ранжування та рекомендації:

- Використовуючи алгоритми ранжування, такі системи забезпечують релевантність результатів.

- Інтеграція з рекомендательними системами допомагає пропонувати альтернативи чи оптимальні рішення.

6. Інтерактивна взаємодія:

- Інтерфейси пошукових систем включають візуалізацію даних, інструменти для роботи з графіками, картами або інтерактивними панелями.

- Це сприяє кращому розумінню отриманих результатів та полегшує прийняття рішень.

Значення пошукових адаптивних веб-систем у СППР

1. Ефективність прийняття рішень:

- Пошукові системи прискорюють доступ до критичної інформації, дозволяючи ухвалювати рішення вчасно.

- Вони значно знижують ймовірність помилок, надаючи користувачам точні й актуальні дані.

2. Масштабованість аналізу:

- Завдяки роботі з великими обсягами даних (Big Data) пошукові системи дозволяють масштабувати аналіз без значних витрат.

- Вони корисні для глобальних компаній, які потребують узгодження інформації з багатьох джерел.

3. Зменшення когнітивного навантаження:

- Автоматизація пошуку та аналізу зменшує обсяг роботи, яку потрібно виконати людині.

- Інтелектуальні системи фільтрують шум, зосереджуючи увагу користувача на важливій інформації.

4. Адаптивність до змін середовища:

- Пошукові адаптивні системи здатні швидко реагувати на зміни в бізнес-середовищі або ринкових умовах.

- Вони динамічно оновлюють свої алгоритми, забезпечуючи актуальність інформації.

5. Підтримка колаборації:

- Системи можуть бути інтегровані в корпоративні середовища, дозволяючи командам спільно працювати над аналізом даних.

- Це сприяє узгодженості рішень у великих організаціях.

6. Інтеграція з іншими інструментами:

- Багато пошукових адаптивних систем інтегруються з CRM-системами, ERP-системами та іншими платформами.

- Це забезпечує комплексний підхід до аналізу даних та прийняття рішень.

Приклади використання.

1. Бізнес-аналітика: Використання Google BigQuery або IBM Watson для аналізу трендів ринку, поведінки клієнтів та прогнозування продажів.

2. Медицина: Інтелектуальні пошукові системи для аналізу наукових досліджень, діагностики хвороб (наприклад, Elsevier ClinicalKey).

3. Освіта: Платформи, як-от Wolfram Alpha, що надають інтерактивні рішення для складних математичних задач.

4. Кібербезпека: ElasticSearch використовується для аналізу логів та виявлення аномалій у реальному часі.

Висновки до першого розділу. В даному розділі розглянуті основні **о**собливості предметно–орієнтованих експертних систем та їх застосування в системах підтримки прийняття рішень. Проведено класифікацію пошукових задач та засобів і технологій інформаційного пошуку в предметно – орієнтованих експертних системах. Визначено значення пошукових адаптивних WEB – систем в СППР та наведено список завдань інформаційного пошуку.

Зазначено, що для розв'язання типових задач, орієнтованих на конкретну предметну область, і які важко розв'язувати за допомогою традиційних математичних методів, необхідно розробляти методи та засоби експертних систем з використанням новітніх інформаційних технологій.

2 АНАЛІЗ СУЧАСНИХ ПОШУКОВИХ СИСТЕМ ТА ЇХ ХАРАКТЕРИСТИКИ

2.1 Основні особливості та функції структурної організації пошукових систем.

Типова схема інформаційно-пошукової системи наведена на рисунку 2.1

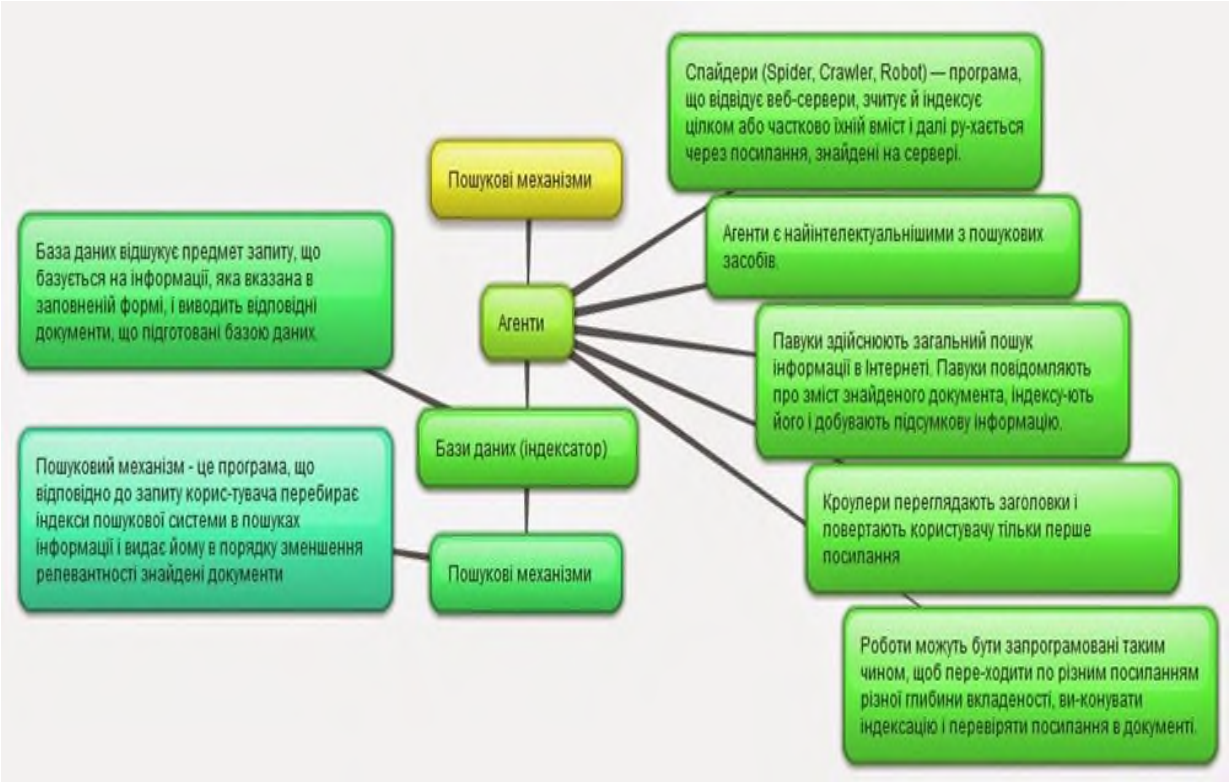

Рисунок 2. 1 - Типова схема інформаційно-пошукової системи в складі експертної системи.

До складу експертної системи входять наступні компоненти (рис. 2.1) [6]:

- БЗ - це ядро експертної системи, є сукупністю знань у формі зрозумілої експертові і користувачеві мови, призначеної для зберігання експертних знань, які використовуються при рішенні задач;

- БД, призначеної для зберігання фактів або гіпотез, які є проміжними рішеннями або результатом спілкування системи із зовнішнім середовищем;

- машини логічного виведення - механізму, що моделює хід міркувань експерта, оперуючи знаннями та даними з метою отримання нових даних зі знань та інших даних, що містяться в робочій пам'яті;

- інтерфейсу користувача, призначеного для ведення діалогу з користувачами для отримання фактів, необхідних для процесу міркування;

- інформаційно-пошукова система;

- підсистеми пояснень, що дає користувачеві можливість розуміти процес отримання результату;

- підсистеми здобуття знань, призначеної для коригування і поповнення БЗ. у діалоговому режимі.

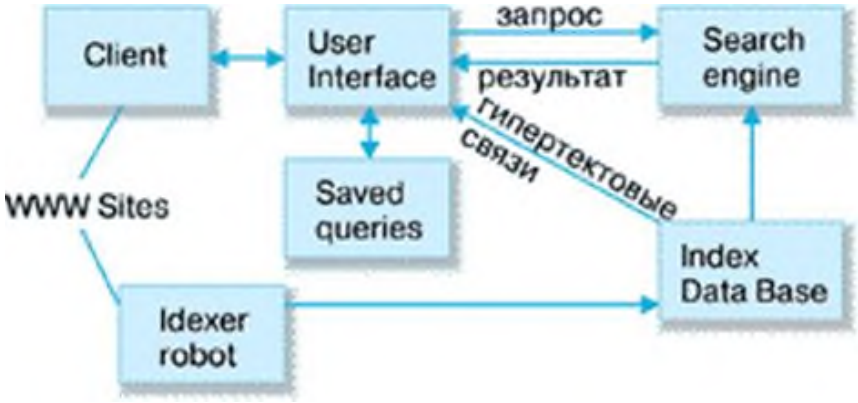

Рисунок 2. 2 - Типова схема інформаційно-пошукової системи.

За допомогою каталогів, можна знайти інформацію або потрібний сайт, розташований у тематичній рубриці. Інший спосіб більш поширений це - посилання (гіпертекстові). І третій спосіб це пошук за допомогою певного слова набраного в стоці пошукової системи, більш звично пошук за ключовими словами в індексі пошуковика [7].

Складання індексу (основні принципи).

База пошукових систем складається з безлічі адрес (сайтів і їх сторінок), є форма з додавання нових сайтів, періодично робот пошукової машини обходить базу адрес для того, щоб оновити індекс, якісь сторінки випадають,

додаються нові, оновлюються старі. Ще цей процес називається апдейт або індексація. Важливою умовою є наявність посилання на сторінку.

Існують кілька роботів або програм, які беруть участь в цьому процесі. Пошуковий "павук" викачує весь текст з доданих сторінок і передає інформацію індексному роботу, що наводить порядок - має в своєму розпорядженні текст в алфавітному порядку, нумерує їх і витягує слова, додаючи їх до індексу і видаляючи зайве. Є "швидкий" робот для індексації часто змінюються сторінок.

І в кінцевому підсумку користувачі вводять в стоці запиту потрібне ключове слово, і програма обробник звертається до індексу і витягує найбільш підходящий список сайтів. Можливе використання розширеного пошуку за допомогою мови запитів, що включає різні оператори умови.

Пристрій індексу.

Сторінки очищаються від мови розміток (тегів) HTML, чистий текст розташовується в алфавітному порядку згідно з правилами машинної морфології, швидше за все робот обрізає слова, залишаючи їх початкову форму. У разі нових або незнайомих слів, а також друкарських помилок вони зберігаються в індексі "як є". Можливо дані розташовуються у таблиці. Більш поширений "координатний" індекс, що враховує місце розташування слів на сторінці.

Крім того існує "прямий" індекс - стисла копія сторінки, що дозволяє відновлювати сторінки.

Правила індексації.

Не індексуються - службові символи, прогалини, теги, знаки пунктуації та зображення, сценарії JavaScript, запити до баз даних. Індексуються - стоп-слова (сполучники, прийменники, цифри, вигуки), будь-які комбінації слів і цифр, посилання і Flash (тобто текст прихований в ісходнике), документи у форматах Word, Exel і PDF. У деяких пошукових індексуються тільки україномовні сайти.

Релевантність.

Пошуковик не може знати, що на думці у користувача, який вводить запит, але завдяки точному словосполученню і фраз, можна знайти, що стосується справи сторінку.

Можна класифікувати запити - за конкретного місця, інформації (телефон, назва, більш точний опис), послуг, транзакційні (купити), загальні запити.

Повнота пошуку - кількість знайдених сторінок, включає різні варіанти відповідей. Точність - кількість відносяться до справи сторінок, розташованих у правильному порядку. З огляду на зміст сайту і його авторитетність. Найбільш відомі мірки це Тематичний індекс цитування (*ТІЦ*), для всього сайту і Page Rank, для кожної сторінки. Ще завдяки посиланням з певним словосполученням з інших сайтів, можливо, підняти сторінку у видачі на більш релевантну, навіть якщо на самому сайті не зустрічається цей вислів, це називається посилальне ранжування.

2.2 Аналіз сучасних пошукових систем та їх характеристики.

Пошукові системи для підтримки прийняття рішень є важливими інструментами в умовах швидкого зростання обсягів даних. Вони дозволяють аналізувати великі обсяги інформації, витягувати корисні знання та робити прогнози. Розглянемо ряд ключових сучасних пошукованих веб-систем, їх особливостей та застосувань представлених на рисунку 2.3.

Рисунок 2.3 – Сучасниі пошукові веб-системи.

Пошукова система Google [8].

- Висока швидкість і релевантність результатів за допомогою алгоритмів розташування (наприклад, PageRank).

- Використання штучного інтелекту (ШІ) та машинного навчання для персоналізації.

- Інтеграція з Google Trends для аналізу трендів і прогнозів.

Підтримка прийняття рішень: отримати актуальні дані, проаналізувати тенденції, розмістити ресурси для стратегічних рішень.

Google (Гугл) – єдина компанія, що зосередила свої зусилля на розробці "ідеальної пошукової системи", яка, за словами співзасновника компанії Леррі Пейджа (Larry Page), "точно визначить, що має на увазі користувач, і покаже саме ті результати, які йому потрібні". З цією метою Google шукає нові методи і відмовляється коритись обмеженням існуючих технологій. У результаті Google розробила власну інфраструктуру і революційну технологію PageRank, яка змінила підхід до виконання пошуку.

Розробники Google розуміли, що для швидшого отримання найбільш точних результатів необхідний новий спосіб налаштування сервера. Більшість пошукових систем використовували декілька великих серверів, які часто працювали поволі при пікових навантаженнях. Компанія Google задіювала зв'язані ПК, такі, що дозволяють швидко знаходити відповіді на всі запити. Впровадження цієї інноваційної технології привело до скорочення часу відгуку, підвищення масштабованості і зниження витрат. З тих пір решта всіх компаній копіює цю ідею, тоді як Google продовжує постійно покращувати внутрішню технологію з метою підвищення її ефективності.

Програмне забезпечення, використовуване для реалізації технології пошуку Google, проводить ряд одночасних обчислень, які займають не більше частки секунди. Традиційні пошукові системи більшою мірою грунтуються на тому, як часто слово з'являється на веб-сторінці. Google же вивчає всю структуру веб-посилань і визначає, які сторінки найбільш важливі, за допомогою PageRank (PR). Потім проводиться аналіз відповідності гіпертексту і вибір сторінок, найбільш відповідних для конкретного пошуку.

На підставі загальної значимості і відповідності запиту Google відображає в першу чергу найбільш релевантні і достовірні результати.

Технологія PageRank об'єктивно оцінює значимість веб-сторінок, грунтуючись на рівнянні, що включає більше 500 мільйонів змінних і 2 мільярдів термінів. Замість того, щоб підраховувати прямі посилання, PageRank розглядає посилання із сторінки А на сторінку Б як голос на користь сторінки Б від сторінки А. Далі по кількості отриманих голосів PageRank визначає значимість даної сторінки.

PageRank також оцінює важливість кожної сторінки, що бере участь в голосуванні. При отриманні голосів від сторінок з більшою значущістю посилання стає ціннішим. Значущі сторінки отримують вищий рейтинг PageRank і відображаються на початку результатів пошуку. Технологія Google використовує сукупні інтелектуальні веб-засоби, щоб визначити значимість сторінки. Людський чинник або підтасовування результатів неможливі, і саме тому користувачі довіряють Google як джерелу об'єктивної інформації, в результатах пошуку якого відсутні профінансовані рекламні оголошення.

Пошукова система Google, як і інші системи, також аналізує зміст сторінки. Проте замість простого сканування тексту сторінки (який може виконати веб-розробник за допомогою метатегів) технологія Google аналізує весь зміст сторінки, особливості шрифтів, розбиття тексту і точне розташування кожного слова. Google також аналізує зміст сусідніх веб-сторінок, щоб переконатися в тому, що отримані результати найточніше відповідають запиту користувача.

Необхідно відзначити наступне. Google аналізує і запам'ятовує:

– як часто мінялися сторінки;

– істотність цих змін;

– як міняється щільність ключових слів на сторінці;

– чи змінювалися якірні тексти посилань.

Домен, на якому знаходиться сайт, є дуже важливим для Google – тому, негативна інформація про домен може ускладнити підвищення рейтингу сайта (причому дуже істотно). При виборі домена необхідно звертати увагу не тільки

на його назву, але і на інформацію про колишнього власника (якщо такий, звичайно, був). Так само, по (поки) до кінця не перевіреними даними, факт покупки домена на короткий термін, є підставою для того, щоб Google почав відноситися до нього з «підозрою». Зазвичай, саме дорвейщики, спамери і ті, хто не збираються займатися розвитком сайту, заводять домени на короткі терміни. Тому необхідно реєструвати домен з таким розрахунком, щоб система не запідозрила у вищеназваних речах (термін більше року, буде цілком прийнятний).

Репутація хостинг-компании – ще один аспект, який враховує Google. Існують "білі" хостинг компанії, які визнані в світі, а є і такі, які підпадають під категорію "сірих". Наприклад, є хостери, які готові без щонайменших перешкод розміщувати у себе дорвеї або ж заборонені сайти. До таких хостерів Google застосовує санкції і, якщо сайт знаходиться на неблагонаійному хості, то ці санкції автоматично торкнуться і власник сайту.

Як було сказано вище, Google оперує значеннями Page Rank для того, що б визначити наскільки вагома сторінка і, яке місце вона займе в результаті. Тому, у зв'язку з цим, необхідно пам’ятати:

– не бажано дуже часто посилатися на сайти, що мають нижчий PR оскільки подібна старанність може бути розцінене, як спам-метод підвищення рівня релевантності. Це не означає, що не можна посилатися на молоді сайти, які ще не встигли стати досить популярними.

– не можна посилатися на сайти у яких PR рівний 0, оскільки це саме ті сайти, які знаходяться в «чорному списку» Google (вони або оштрафовані, або заборонені).

Для повного розуміння, що являє собою PageRank, необхідно знати декілька фактів:

– PageRank – це число, що характеризує виключно голосуючу здібність всіх вхідних посилань на сторінку і те, як сильно вони рекомендують цю сторінку.

– Кожна унікальна сторінка сайту, проіндексована Google, має вагу PageRank. Помилково вважати, що вага головної сторінки сайту є вагою всього сайту в цілому.

– Внутрішні посилання сайту враховуються при розрахунку ваги PageRank для інших сторінок сайту.

– PageRank незалежний, він не бере до уваги текст посилань. Звичайно, вони зв'язані, але це не одне і те ж.

Математична модель, що описує PageRank, виглядає таким чином.

$$PR(A) = (1-d) + d\left(\frac{PR(T_1)}{C(T_1)}\right) + \left(\frac{PR(T_2)}{C(T_2)}\right) + ... + \left(\frac{PR(T_n)}{C(T_n)}\right) \qquad (2.1)$$

де $PR(A)$ – це PageRank сторінки A (та вага, яку ми хочемо обчислити)

d – коефіцієнт затухання, який зазвичай приймається рівним 0.85-0.9. Виражає вірогідність того, що користувач, що зайшов на сторінку, продовжуватиме далі переходити по посиланнях;

n – кількість сторінок, що посилаються на поточну сторінку;

T_i – і-та сторінка, що посилається;

C – кількість зовнішніх посилань на сторінці.

Однак на PageRank накладено обмеження:

$$\sum_{k=1..N} PR_k = N \qquad (2.2)$$

де N – загальна кількість веб-сторінок в мережі Інтернет

Тобто, середній PageRank рівний одиниці. Обмеження це витікає з нормування ймовірності перебування користувача по всій мережі – сума ймовірності по всіх сторінках рівна одиниці. Таким чином

$$Ймовірність_i = \frac{PageRank}{число\ сторінок\ в\ мережі}$$

Оскільки сторінок, що посилаються, може бути багато, і загальна кількість сторінок в пошуковій системі Google достатньо велика (близько десятка більйонів штук) а також їх кількість постійно росте, то представляти вагу сторінки в абсолютних значеннях було б неправильно. Для цього ввели поняття TLPR — ToolBar PageRank (тулбарний PageRank), який має значення від 0 до 10.

Для того, щоб укласти всі ваги сторінок між значеннями від 0 до 10 використовують логарифмічну шкалу.

$$TLPR = Log_{base}(PR) \cdot a$$

де $base$ – основа логарифма, яка залежить від кількості сторінок в пошуковій машині (можливо і від ряду інших чинників). Зазвичай його приймають рівним 7;

a – коефіцієнт приведення, який задовольняє нерівності $0 < a \le 1$. Оптимізаторам його можна прийняти рівним одиниці для спрощення розрахунків.

З вищесказаного невірно робити висновки, що нульовий TLPR означає нульовий реальний PageRank. По математичній моделі PR видно, що навіть при $n = 0$, ми отримаємо мінімальний $PR_{\min} = (1-d) = 0.15$. Це значення відповідає $TLPR \approx -1$. При таких (мінусових) значеннях тулбарного PR вважається що $PR = N / A$ (ще не визначений), проте він також робить вплив на розподіл ваги між посиланнями.

Виходячи з принципів розрахунку Google PageRank, можна тепер легко розрахувати, з яких посилань потрібно посилатися і скільки потрібно посилань, щоб отримати той або інший PR. Також можна прогнозувати PR. Один з важливих висновків, який можна зробити з вищесказаного полягає в наступному. Якщо маємо новий сайт із сторінками більше 10,000 (число сторінок залежить від кількості посилань з них на інші сторінки), вони правильно пролінковані і кожна посилається на головну сторінку, то головна

сторінка отримає хорошу вагу від цих посилань. Математично можна зобразити так:

$$PR_{\min} = (1-d) = 0.15;$$
$$PR = 0.15 + 0.85 \cdot \frac{20000}{10}$$

(за умови, якщо в середньому 10 посилань на сторінці)

$$TLPR = Log_{base}(1700.15, 7) = 3.823 \approx 4$$

Яскравий приклад високого PR без єдиного зовнішнього посилання з інших сайтів.

IBM Watson Discovery.

Сфера застосування: корпоративний пошук і аналіз даних.

Особливості:

- Використання природної мови (NLP) для розуміння запитів.

- Автоматизація обробки великих масивів даних.

- Підтримка роботи з неструктурованими даними (наприклад, текстами, PDF-файлами).

Підтримка прийняття рішень: створення комплексних звітів, автоматизований аналіз текстів для виявлення ключових ідей.

Wolfram Alpha.

Сфера застосування: рішення задач і аналітика в наукових, технічних та бізнес-сферах.

Особливості:

- Пошук не просто інформації, а готових відповідей.

- Достатня математична база для виконання розрахунків.

- Можливості моделювання, статистичного аналізу та прогнозів побудови.

Підтримка прийняття рішень: моделювання складних сценаріїв, аналіз даних у технічних проектах.

DuckDuckGo.

Сфера застосування: пошук з високим рівнем конфіденційності.

Особливості:

- Не збирає особисті дані користувачів.

- Мінімалістичний інтерфейс і точні результати.

Підтримка прийняття рішення: підходить для збору інформації без ризику витоку даних.

2.3 Аналіз проблем та постановка задачі побудови пошукової системи.

На підставі результатів проведеного аналізу можна зробити висновок, що для того, щоб пошукова система могла бути коректно вбудована в експертну систему потрібно змоделювати і розробити адаптивний програмний код, написаний за допомогою об’єктно-орієнтованого програмування. Мовою для програмування доцільно використати PHP.

Для вирішення задачі ранжування, а також для оцінки релевантності пошукових документів запиту, в умовах неоднозначності, доцільно використовувати знання експертів-фахівців та методи теорії ймовірності.

Виходячи з цього можна з формулювати постановку задачі розробки системи пошуку та оцінки релевантних документів при побудові предметно–орієнтованої експертної системи з використанням WEB - технологій.

Для цього необхідно:

- Провести аналіз сучасних підходів до побудови предметно–орієнтованих експертних систем.

- Визначити основні фактори впливу на процес пошуку інформації.

- Дослідити та провести оцінку основних факторів впливу по ступеню важливості використовуючи знання експертів-фахівців.

- Розробити математичну модель процесу пошуку інформації та провести оцінку міри релевантності документу запиту користувача з застосуванням ймовірнісного методу Байесовської мережі довіри.

- Розробити програмно-інструментальні засоби для реалізації системи пошуку та оцінки релевантних документів при побудові предметно–орієнтованої експертної системи з використанням WEB - технологій.побудови пошукової системи.

Висновки до другого розділу. В даному розділі проаналізовані основні особливості та функції структурної організації пошукових систем. Розглянуто типові схеми інформаційно-пошукової системи в складі експертної системи. та їх характеристики. Проведений аналіз проблем до побудови сучасних пошукових систем.

На підставі результатів проведеного аналізу зазначено, що для вирішення задачі ранжування інформаційних запитів, а також для оцінки релевантності пошукових документів запиту, в умовах неоднозначності, доцільно використати знання експертів-фахівців та методи теорії ймовірності

Виходячи з цього була з формульована постановка задачі на розробку системи пошуку та оцінки релевантних документів при побудові предметно–орієнтованої експертної системи з використанням WEB - технологій.

РОЗДІЛ 3. РОЗРОБКА МАТЕМАТИЧНОЇ МОДЕЛІ ПРОЦЕСУ ПОШУКУ ТА ОЦІНКИ РЕЛЕВАНТНИХ ДОКУМЕНТІВ З WEB СЕРЕДОВИЩА

3.1 Розробка математичної моделі пошукової системи.

Нехай є деяка кількість документів, які були отриманні з мережі Інтернет. Кожен документ має певні характеристики, з різноманітними реквізитами, притаманними певним документам. В пошуковій системі задаються потрібні ключові слова, які характеризують саме ту інформацію, яку потрібно знайти. Сукупність ключових слів та критеріїв називається пошуковим запитом [9].

Ступінь відповідності кожного конкретного документу запиту називається релевантністю документу запиту. Задача пошукової системи – представлення користувачеві максимально релевантних результатів, які максимально відповідають його запиту.

Таким чином, необхідно розробити математичну модель інтелектуальної системи, яка дозволить оцінити міру релевантності кожного документу заданого запиту. Мірою релевантності назвемо число, у відповідності з його значенням за яким можна відсортувати документи за релевантністю. Це число повинно мати наступні параметри:

- невід’ємне дійсне число;
- міра релевантності буде настільки вищою, наскільки буде вища сама релевантність запиту;
- обмеження міри релевантності зверху.

Для моделювання використаємо ймовірнісний метод, Байєсовську мережу довіри, яка використовується для моделювання предметних областей, котрі характеризуються невизначеністю [10]. Ця невизначеність може бути обумовлена недостатнім розумінням предметної області або комбінацією заданих факторів.

Байєсівські мережі також називають баєсівськими мережами довіри (БМД) або просто мережами довіри. БМД – це граф, вершини якого з’єднані направляючими ребрами, з заданою кожному вузлу ймовірнісною функцією.

Мережа у БМД представляє собою напрямлений ациклічний граф, у якому немає направленого маршруту, який починається і закінчується у одній і тій же вершині.

Якщо у вершині не існує ребер, спрямованих до неї, вона буде містити таблицю безумовних ймовірностей своїх станів. У разі дискретної вершини така таблиця містить розподіл ймовірностей між усіма можливими станами цієї вершини. Якщо ж у є батьки (одне або кілька ребер, спрямованих до неї), то така вершина містить таблицю умовних ймовірностей (CPT, conditional probability table), кожен осередок якої містить умовну ймовірність перебування вершини в певному стані для випадку певної конфігурації станів всіх її батьків. Таким чином, кількість клітинок у таблиці умовних ймовірностей дискретної вершини БСД дорівнює добутку кількості можливих станів цієї вершини на твір кількості можливих станів всіх її батьківських вершин.

Тривіальна БСД, показана на рис. 3.1, відображає причинно-наслідковий взаємозв'язок між двома елементами деякої предметної області - A і B. Наявність причинно-наслідкового зв'язку від A до B означає нашу думку про те, що якщо A знаходиться в деякому стані, то це впливає на стан B.

Випадкова дискретна змінна, яку представляє вершина A, може знаходиться в одному з двох станів, як показано в таблиці 3.1, - a1 або a2. Вершина B має три можливих стани: b1, b2, b3.

Таблиця 3.1 - умовна ймовірностей для вершин БМД

B	P(bi|a1)	P(bi|a2)
b1	1	0.6
b2	0	0.2
b3	0	0.2

Таблиця 3.2 - умовна ймовірностей для вершин БМД

A	P(ai)
a1	0.5
a2	0.5

Оскільки вершина А не має батьків, значення ймовірностей її станів не є залежними (про розподіл ймовірностей вершини А в такому випадку свідчать, що задані апріорні ймовірності її станів). Для вершини В, навпаки, ймовірність станів залежать від стану її батьківської вершини А:

- Якщо А знаходиться в стані a1, то В знаходиться в стані b1;

- Якщо А знаходиться в стані a2, то ймовірність того, що В знаходиться в стані b1 дорівнює 0.6, а в станах b2 і b3 - 0,2.

Якщо вершина А не має батьків, то замість умовних ймовірностей (автоматично) використовуються безумовні ймовірності Р (А).

Саме процес обчислення ймовірностей є основою для прийняття рішень в умовах невизначеності на основі байєсівських мереж [11]. Розкриття невизначеності (dealing with uncertainty) здійснюється в БСД шляхом обчислення ймовірностей станів цікавлять нас вершин на основі наявної інформації про значення (частини) інших вершин мережі. Математичний базис для цього процесу визначає байєсовський підхід до аналізу невизначеності і відповідний йому апарат класичної теорії ймовірностей.

Вершина БМД представляє собою основу байєсівського підходу складає поняття умовної ймовірності $P(A|B)=x$, яка означає, що за умови виникнення В (і всього іншого, що не має відношення до В) ймовірність виникнення А дорівнює х. Спільна ймовірність настання А і В визначається формулою повної ймовірності

$$P(A,B)=P(A|B)P(B)=P(B|A)P(A)\,. \qquad (3.1)$$

Рівняння (3.1) фундаментальний принцип обчислення ймовірностей і основа для теореми Байєса:

$$P(B \mid A) = \frac{P(A \mid B)P(B)}{P(A)}. \qquad (3.2)$$

Теорема Байєса застосовується, якщо в даному розпорядженні є інформація про залежних змінних, а суть дослідження полягає у визначенні ймовірності вихідних змінних [12]. Так, нехай відома умовна ймовірність $P(B \mid A)$ виникнення деякої події В за умови, що має місце подія А. Тоді теорема Байєса дає рішення зворотного завдання - яка ймовірність виникнення події А, сталася подія В.

Нехай $A_1 A_2,..., A_n$ - повна група несумісних взаємовиключних подій (або альтернативних гіпотез). Тоді апостеріорна ймовірність $P(A_j \mid B)$ кожного з подій A_j, $j = 1..n$ за умови, що відбулася подія В, виражається апріорною ймовірністю: A_j

$$P(A_j \mid B) = \frac{P(B \mid A_j)P(A_j)}{P(B)} = \frac{P(B \mid A_j)P(A_j)}{\sum_{j=1}^{n} P(B \mid A_j)P(A_j)}. \qquad (3.3)$$

Оцінка релевантності документів.

Одне із завдань, в якій успішно застосовуються байєсовські мережі - завдання класифікації. Так званий наївно-байєсовський класифікатор, який представляє собою просту байєсівську мережу, є одним з найефективніших класифікаторів. Цей підхід передбачає, що завдання оцінки релевантності також можна розглядати як завдання класифікації. Кожен документ належить до однієї з двох непересічних областей: С1 - релевантні документи, С2 нерелевантні документи.

У такому випадку, задача оцінки релевантності документа запиту представляється у вигляді задачі віднесення його до одного з двох класів. У

цьому випадку, приналежність документа до першого класу дозволяє свідчити, що цей документ є релевантним запиту.

Вирішимо цю задачу за допомогою байесовской мережі, зіставивши поняттю «документ» вершину мережі. Ця вершина може знаходитися в двох станах: c1 - «документ релевантний» і c2 - «документ не релевантний». Апріорні ймовірності цих станів покладемо рівними 0.5, що відповідає поняттю невизначеності в ймовірнісному аналізі:

$$P(c_1) = P(c_2) = 0.5$$

Якщо в результаті обчислень отримаємо, що ймовірність знаходження цього вузла в стані «документ релевантний» дорівнює 0.9, то це означатиме, що з імовірністю 0.9 даний документ належить до класу C1.

Далі, нехай $F = \{F_i\}, i = 1..n$ - безліч факторів, що впливають на релевантність документа. Розглянемо, наприклад, такий фактор, як наявність ключового слова запиту в заголовку документа. Очевидно, що наявність ключового слова в заголовку підвищує релевантність документа. Тоді введемо в мережу вершину F1, відповідну події «ключове слово в заголовку документа». Ця вершина буде мати два стани:

f11 - «ключове слово зустрілося в заголовку документа»

f12 - «ключове слово не зустрінеш в заголовку документа».

Умовні ймовірності $P(f_{1j} \mid c_i),\ i, j = 1..2$, то для цього є таблиця умовних ймовірностей для вершини F1, і можна розрахувати ймовірності $P(c_i \mid f_{1j}),\ i, j = 1..2$.

Для віднесення документа D до класу релевантних у разі, коли відомо стан f_{1j}, використовується очевидне правило: якщо $P(c_1 \mid f_{1j}) > P(c_2 \mid f_{1j})$, то $D \in C_1$.

Отже, для визначення релевантності повинні бути виділені всі фактори, складові безліч F, і задані таблиці умовних ймовірностей для кожного фактору.

Кожний з факторів розраховується відповідним чином для кожного ключового слова запиту.

Таким чином, вершини мережі в даному випадку - це фактори, що впливають на ймовірність даного «головного» вузла, що відповідає за релевантність документа в цілому. Байєсова мережа для даної задачі має вигляд.

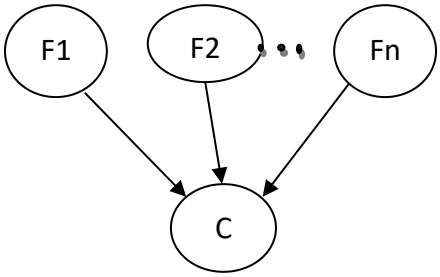

Рисунок 3.1 – Байєсівська мережа для оцінки релевантності документа запиту.

C - вершина мережі, що представляє собою ймовірність того, що документ релевантний запитом, а F1, F2 … Fn – фактори, що враховуються при розрахунку цієї ймовірності. Суттєвим моментом є напрямок причинно-наслідкових зв'язків у мережі. Так, стрілки виходять з вершини C і входять до вершини Fi. Тут байєсовська мережа виконує зворотний логічний висновок - визначає ймовірність кожного стану вершини C при відомих станах вершин Fi.

Вищерозглянутий випадок фактору F1 брав одне з двох значень. Однак фактори можуть мати різну природу - вони можуть приймати кілька значень, а можуть взагалі не бути дискретними. У загальному випадку розглядаємо певний діапазон зміни значень кожного фактору. Нехай деякий фактор Fi приймає значення Тоді нормується значення цього чинника в діапазон [-1; 1] за допомогою формули

$$\tilde{x} = \frac{x - \frac{x_{\min} + x_{\max}}{2}}{x_{\max} - \frac{x_{\min} + x_{\max}}{2}} \quad (3.4$$

і приймаються розрахункові ймовірності для відповідної вершини рівними

$$\tilde{P}(f_i \mid c_1) = \frac{1 - [1 - 2 \cdot P(f_i \mid c_1)] \cdot \tilde{x}}{2},\ i = 1..n$$
$$\tilde{P}(f_i \mid c_2) = \frac{1 - [1 - 2 \cdot P(f_i \mid c_2)] \cdot \tilde{x}}{2},\ i = 1..n, \quad (3.5)$$

де $P(f_i \mid c_1)$ - елемент таблиці умовних ймовірностей для і-ої вершини мережі, що показує, з якою ймовірністю в релевантному документі фактор Fi приймає максимальне значення $x = x_{\max}$; $P(f_i \mid c_2)$ – ймовірність, з якою фактор Fi приймає максимальне значення $x = x_{\max}$ у нерелевантному документі.

Отримані розрахункові ймовірності $\tilde{P}(f_i \mid c_1)$ і $\tilde{P}(f_i \mid c_2)$ тепер можна використовувати у формулі Байєса:

$$P(c_1 \mid f_i) = \frac{P(f_i \mid c_1) \cdot P(c_1)}{P(f_i \mid c_1) \cdot P(c_1) + P(f_i \mid c_2) \cdot P(c_2)},\ i = 1..n. \quad (3.6)$$

Таким чином, вищеописана схема дозволяє врахувати як дискретні, так і безперервні значення факторів, що впливають на загальну релевантність документа. При цьому, якщо зростання значення фактору відповідає зменшенню релевантності (наприклад, кількість днів, що минули з дати публікації оголошення), то достатньо повторити ці міркування для випадку, коли елемент таблиці умовних ймовірностей $P(f_i \mid c_1)$ показує, з якою

ймовірністю в релевантному документі фактор Fi приймає мінімальне значення $x = x_{\min}$.

У даному випадку мережа є досить тривіальною, щоб розрахунок ймовірностей міг бути виконаний послідовним застосуванням теореми Байєса. Такий розрахунок можливий тільки в тому випадку, якщо працює припущення про умовну незалежності вершин мережі. Умовна незалежність вершин байесовской мережі означає блокування впливу між цими вершинами. Змінні (безлічі змінних) F1 і F2 є незалежними при відомому стані змінної A, якщо

$$P(F_1 \mid A) = P(F_1 \mid A, F_2). \qquad (3.7)$$

Це означає, що якщо стан вершини A відомо, то ніяка інформація про F1 не змінює ймовірності F2. Це представляється відсутністю причинно-наслідкових зв'язків між усіма чинниками безлічі F.

Насправді це припущення, очевидно, є абсолютно нереалістичним (саме тому класифікатори подібної структури і носять назву «наївних»). У той же час порушення цього припущення в умовах реального світу не виявляє істотного впливу на кінцевий результат. Виявляється, що такий послідовний підхід є в даному випадку перевагою, оскільки різко знижує обчислювальну складність і відповідно швидкість роботи алгоритму.

Засвідчуючи про отримання чисельних значень для таблиць умовних ймовірностей, слід зазначити, що концептуально для вирішення цього завдання виділяються два підходи [6]:

- отримання інформації від експертів предметної області;
- отримання інформації на підставі експериментальних даних.

Таблиці умовних ймовірностей найчастіше генеруються на підставі даних за допомогою статистичних методів. Проте варто відзначити, що принципово суб'єктивний байєсовський підхід не вимагає «об'єктивності» ймовірностей, а тому дозволяє при формуванні таблиць умовних ймовірностей спиратися на суб'єктивні оцінки експертів. Умовні ймовірності, чисельні значення які

використовуються для розрахунку, отримані на підставі злиття результатів статистичних досліджень та експертних оцінок. Було проведено статистичний аналіз безлічі релевантних і нерелевантних документів для різних запитів за різними джерелами інформації і занесено ці значення до таблиці умовних ймовірностей мережі.

3.2 Оцінка релевантності ранжирування документу в пошуковій системі.

Для оцінки ранжирування в таблиці 3.3 наведені найбільш впливові фактори, що впливають на релевантність документу пошуку, які представлені у вигляді вершин байесовської мережі довіри, кожна з яких може приймати відповідні стани до заданих втаблицях умовних ймовірностей для цих вершин. При надходженні пошукового запиту система виконує розрахунок кожного фактору для кожного ключового слова і виконує поширення відповідних розрахункових ймовірностей у мережі. Результатом роботи є ймовірність $P(C \mid F_1, F_2 ... F_n)$ для кожного наявного документа D, яка і є мірою релевантності документа запиту.

Якщо $P(C = c_1 \mid F_1, F_2 ... F_n) > P(C = c_2 \mid F_1, F_2 ... F_n)$, то документ релевантний запиту, тобто

$$P(C = c_1 \mid F_1, F_2 ... F_n) > 0.5), \; то \; D \in C_1 . \qquad (3.8)$$

Документи, що задовольняють вирішальному правилу (3.8), передаються на підсистему з нормованою мірою релевантності.

$$P = \frac{P(C \mid F_1, F_2 ... F_n) - P_{\min}}{P_{\max} - P_{\min}} \cdot 100\% = 2 \cdot [P(C \mid F_1, F_2 ... F_n) - 0.5] \cdot 100\% \qquad (3.9)$$

Таблиця 3.3 - Фактори включені до байєсівської мережі в якості вершин.

Фактор	**Розрізняючі стани**	**Пояснення**
F1 - Входження ключового слова у **заголовок** документа	1) 0 входжень 2) 1 і більше входжень	Наявність ключового слова в *position title* підвищує релевантність шуканого документа; відсутність знижує релевантність
F2 - Входження ключового слова у **короткий опис** документу	1) 0 входжень 2) 1 входження	Наявність ключового слова в *summary* (перші 25 слів статті) підвищує релевантність статті; відсутність не змінює релевантністю
F3 - Багаторазові входження ключового слова в **короткий опис** документу	1) Менше 2 входжень 2) 2 і більше входжень	Входження ключового слова в *summary* два і більше рази підвищує релевантність статті
F4 - Кількість входжень **ключового слова** в шуканому тексті	1) Менше 2 входжень 2) Від 2 до 7 входжень 3) Більше 7 входжень	Наявність у тексті ключового слова 2 і більше рази підвищує релевантність запиту (нелінійно, по дискретним значенням фактора "2», «3», «4», «5», «6», «7 і більше»); наявність рівно одного входження не змінює релевантністю, відсутність входжень знижує релевантність
F5 - Положення входження ключового слова в тексті документу	Значення в інтервалі від 0.6086 до 1	Положення слова в тексті документа представляється числом від 0 (кінець документа) до 1 (початок документа); більшого значення цього числа підвищує релевантність (нелінійно, по безперервним значенням фактора)
F6 - Кількість входжень **біграм** (пар слів) в заголовок документа	1) 0 входжень 2) 1 і більше входжень	Наявність у *position title* фрази (біграми), що збігається з фразою з двох ключових слів, підвищує релевантність; відсутність фрази не змінює релевантністю

F7 - Кількість входжень **біграм в повний текст** документа пошуку	1) 0 входжень 2) Від 1 до 4 входжень 3) Більше 4 входжень	Наявність в повному тексті статті (*summary* + *text* + *skills*) біграми 1 і більше разів підвищує релевантність документа (нелінійно, по дискретним значенням фактора «1», «2», «3», «4 і більше»); відсутність фрази не змінює релевантності документа
F8 - Значення фактору **TF*IDF** для ключового слова	1) 0 2)Значення в интервалі від 0 до 4 3) Значення більше 4	Більше значення чинника TF * IDF [7], що враховує частоту входження ключового слова (TF) і вага слова в документі (IDF), підвищує релевантність документа (нелінійно, по безперервним значенням фактора в інтервалі від 0 до 4, при значенні «4 і більше »- максимально); значення 0 не змінює релевантності документа
F9 - Дата публикації статті	Значення в інтервалі від 0 до 50 (днів)	Фактор представляє кількість днів, що минув з моменту публікації статті і до поточної (сьогоднішньої) дати. Більше значення чинника знижує релевантність документа (нелінійно, по дискретним значенням «1», «2», ... «49», «50 і більше»); значення 0 («сьогодні») не змінює релевантністю статті

3.3 Моделювання роботи системи на ППЗ Genie 2.0

На основі розрахунків, проведених в підрозділах 3.1 і 3.2 можна побудувати узагальнений граф байесовської мережі довіри для визначення ймовірності нормованої міри релевантності пошукового запиту, який зображений на рисунку 3.2. Пропонується моделювання роботи в програмі Genie 2.0

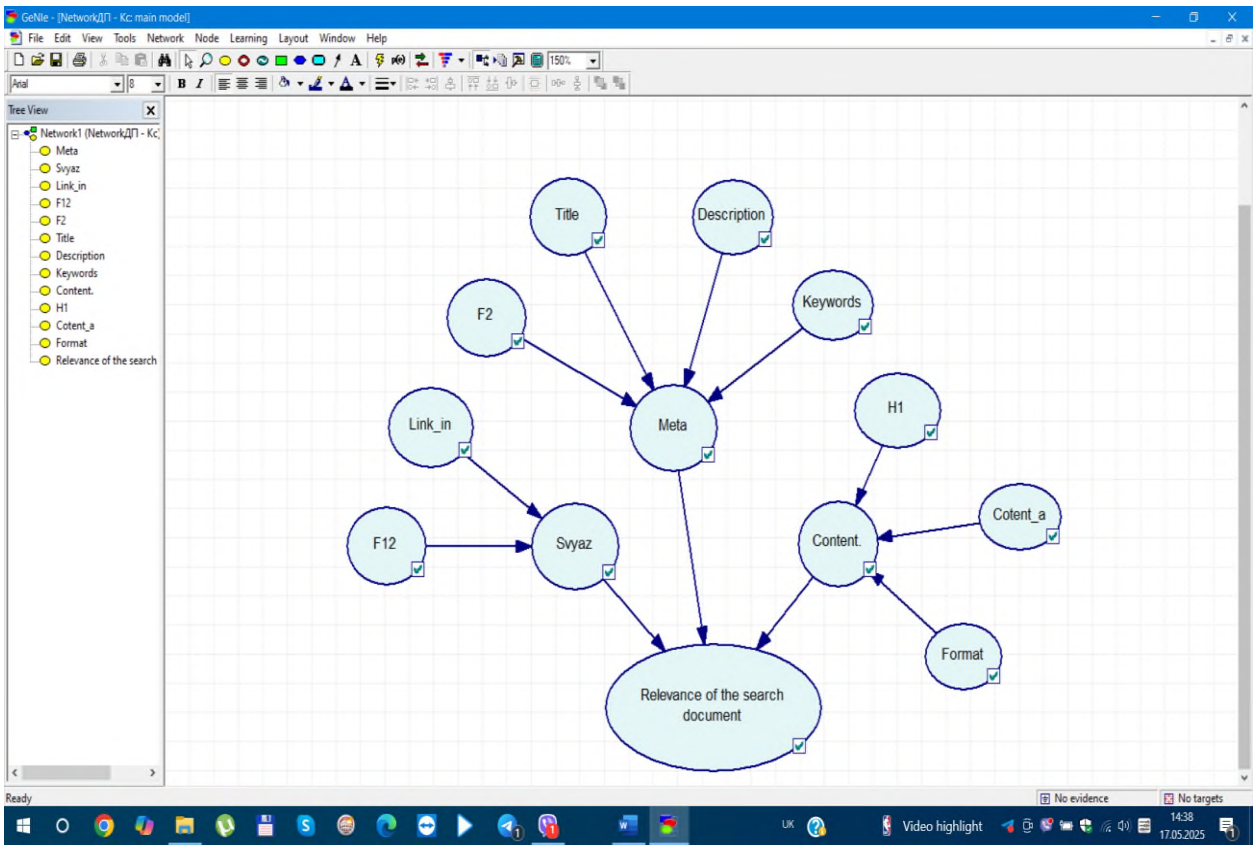

Рисунок 3.2 – Байєсівська мережа оцінки релевантності ранжування документу.

Визначення ймовірності починається з надходження у мережу інформації на вузли зовнішніх факторів та результатів ранжирування на вузли особистісних факторів. Нехай на момент початку розрахунків відомі значення, показані на рисунку 3.3. Після оновлення інформації і обчислення значень вагомості факторів, отримуємо наступні результати, показані на рисунках 3.4 -3.10.

Node Name	State Name	Special Name F...	Special Name	Node Id	State Id	Prior Probability	Cost	Type	Ranked	Mandatory	Target State	De...	No...	Qu...	St...	Tr...	Links
Content				Node11				Auxiliary					Add				Add
	True														Add		Add
	False														Add		Add
Content_a				Node8				Auxiliary					Add				Add
	True					0.28									Add		Add
	False					0.72									Add		Add
Description				Node4				Auxiliary					Add				Add
	True					0.65823									Add		Add
	False					0.34177									Add		Add
Format				Node15				Auxiliary					Add				Add
	True					0.31									Add		Add
	False					0.69									Add		Add
H1				Node7				Auxiliary					Add				Add
	True					0.36									Add		Add
	False					0.64									Add		Add
Keywords				Node5				Auxiliary					Add				Add
	True					0.54									Add		Add
	False					0.46									Add		Add
Link_in				Node9				Auxiliary					Add				Add
	True					0.44									Add		Add
	False					0.56									Add		Add
Meta				Node6				Auxiliary					Add				Add
	True														Add		Add
	False														Add		Add
Rangering				Node13				Auxiliary					Add				Add
	True														Add		Add
	False														Add		Add
Svyaz				Node12				Auxiliary					Add				Add
	True														Add		Add
	False														Add		Add
Title				Node3				Auxiliary					Add				Add
	True					0.72									Add		Add
	False					0.28									Add		Add

Рисунок 3.3 - Заповнення вихідних даних по результатам експертних оцінок.

Встановлення значення вагомості для фактору Title приведена на рисунку 3.4.

True	0.72
False	0.28

Рисунок 3.4 - Значення вагомості для Title

Встановлення значення вагомості для фактору Description показано на рисунку 3.5.

True	0.65823
False	0.34177

Рисунок 3.5 – Значення вагомості для Description

Для встановлення значення вагомості фактору Keywords було зазначено дані, показані на рисунку 3.6.

True	0.54
False	0.46

Рисунок 3.6 - Значення вагомості для Keywords

На рисунку 3.7 показані значення вагомості фактору Link_in.

True	0.44
False	0.56

Рисунок 3.7 - Значення вагомості для Link_in

Встановлення значення вагомості для фактору H1 показано на рисунку 3.8.

True	0.36
False	0.64

Рисунок 3.8 - Значення вагомості для H1

Встановлення значення вагомості фактору Cotent_a було зазначено дані, показані на рисунку 3.9.

True	0.28
False	0.72

Рисунок 3.9 - Значення вагомості Cotent_a

На рисунку 3.10 показані значення вагомості фактору Format.

True	0.31
False	0.69

Рисунок 3.10 – Значення вагомості Format

Встановлення розподілу всіх відношень вузла Meta, який показано на рисунку 3.11. Коефіцієнт ймовірності впливу множини об’єкту Meta на ранжирування веб-сторінки показано на рисунку 3.12.

Parent	Title	Description	Keywords	LEAK
State	True	True	True	
True	0.66	0.49	0.61	0.35
False	0.34	0.51	0.39	0.65

Рисунок 3.11 – Розподіл відношень вузла Meta

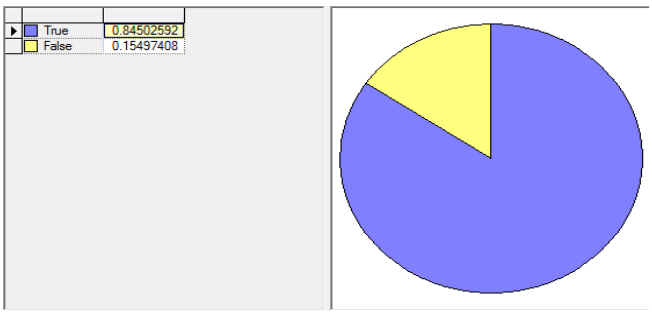

Рисунок 3.12 - Коефіцієнт ймовірності впливу множини об’єкту Meta

Розподіл всіх відношень вузла Content, який показано на рисунку 3.13 і коефіцієнт ймовірності впливу множини об’єкту Content на ранжирування веб-сторінки показано на рисунку 3.14.

Parent	H1	Content_a	Format	LEAK
State	True	True	True	
True	0.32	0.29	0.26	0.45
False	0.68	0.71	0.74	0.55

Рисунок 3.13 - Розподіл відношень вузла Content.

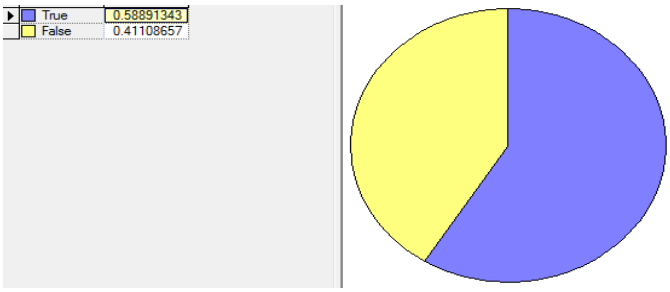

Рисунок 3.14 - Коефіцієнт ймовірності впливу множини об’єкту Content

На рисунку 3.15 показано відношення вузла Svyaz і коефіцієнт ймовірності впливу об’єкту Svyaz на ранжирування веб-сторінки показано на рисунку 3.16.

Parent	Link_in	LEAK
State	True	
True	0.42	0.44
False	0.58	0.56

Рисунок 3.15 – Відношення вузла Svyaz

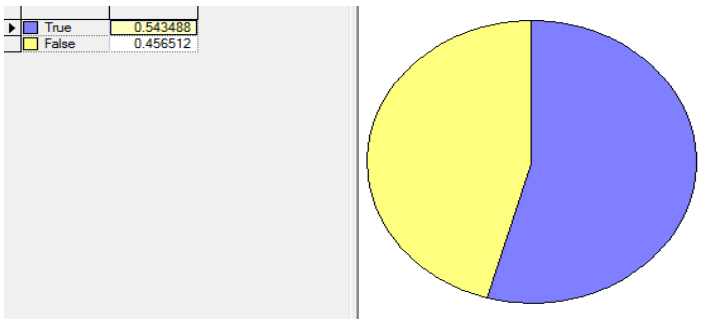

Рисунок 3.16 - Коефіцієнт ймовірності впливу об’єкту Svyaz

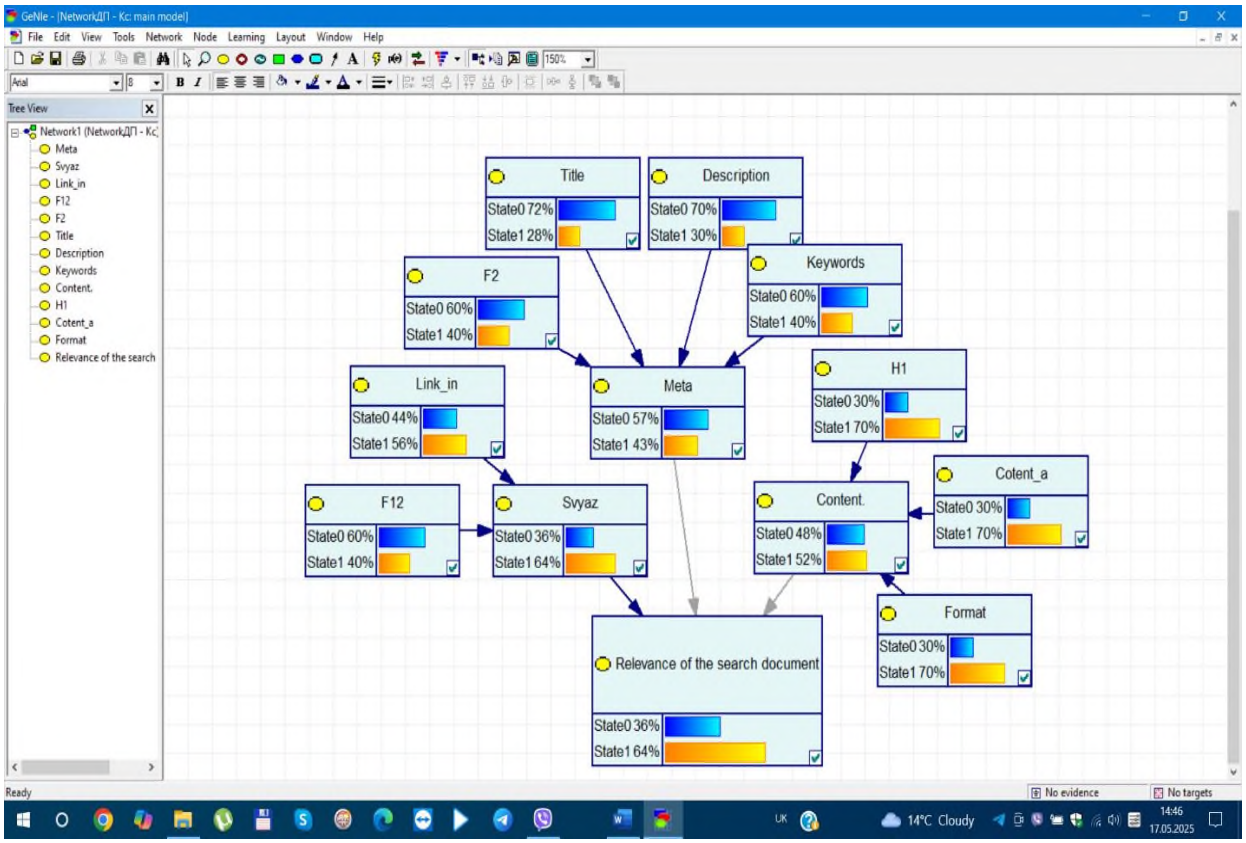

Рисунок 3.17 – Результати оцінки релевантності ранжування документу при меншій кількості збігів факторів

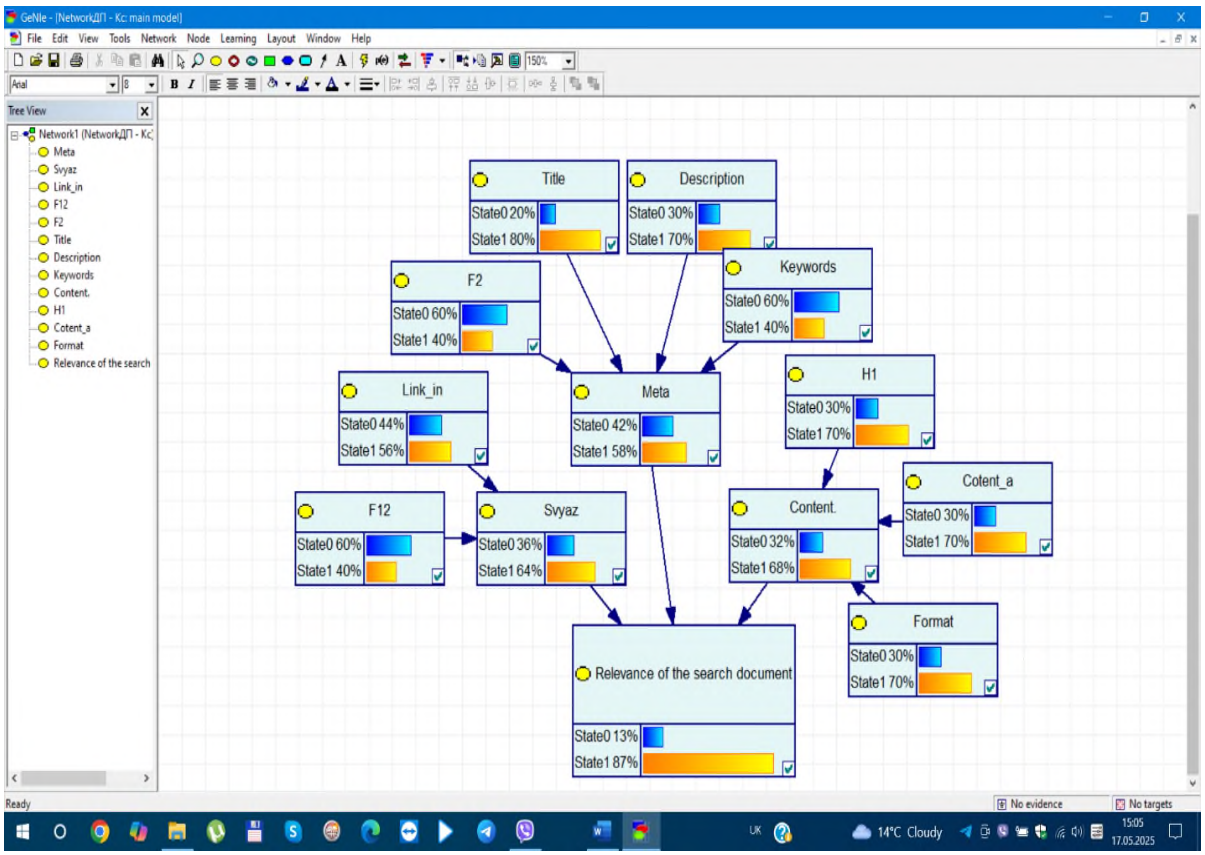

Рисунок 3.18 - Результати оцінки релевантності ранжування документу при більшій кількості збігів факторів.

За результатами моделювання на рисунку 2.18 видно, що релевантність ранжирування документу при більшій кількості збігів факторів – збільшується. Так при показаннях Meta 72%, Content 45% і Svyaz 39% релевантність ранжирування становить 87% Якщо збільшити відсоток факторів то навіть при меншій кількості збігів релевантність значно зростає .

Висновки до третього розділу. В даному розділі наведена розробка математичної моделі інтелектуальної системи, яка дозволяє оцінити міру релевантності кожного документу заданого запиту. Для моделювання використаний ймовірнісний метод, Байєсовську мережу довіри, яка використовується для моделювання предметних областей, котрі характеризуються невизначеністю, що може бути обумовлена недостатнім розумінням предметної області або комбінацією заданих факторів.

В розділі проведена оцінка релевантності ранжирування документу в пошуковій системі з урахуванням найбільш впливових факторів, що впливають на релевантність документу пошуку, значення яких представлені у вигляді вершин байесовської мережі довіри заданих експертами.

На основі проведених розрахунків побудований узагальнений граф байесовської мережі довіри та проведена оцінка в програмному середовищі Genie 2.0.

РОЗДІЛ 4. РОЗРОБКА ПРОГРАМНО – АЛГОРИТМІЧНОГО ЗАБЕЗПЕЧЕННЯ ПОШУКОВОЇ СИСТЕМИ

4.1 Розробка загальної структури пошукової системи та алгоритми її роботи

На рисунку 4.1. представлена розробка загального алгоритму роботи пошукової системи.

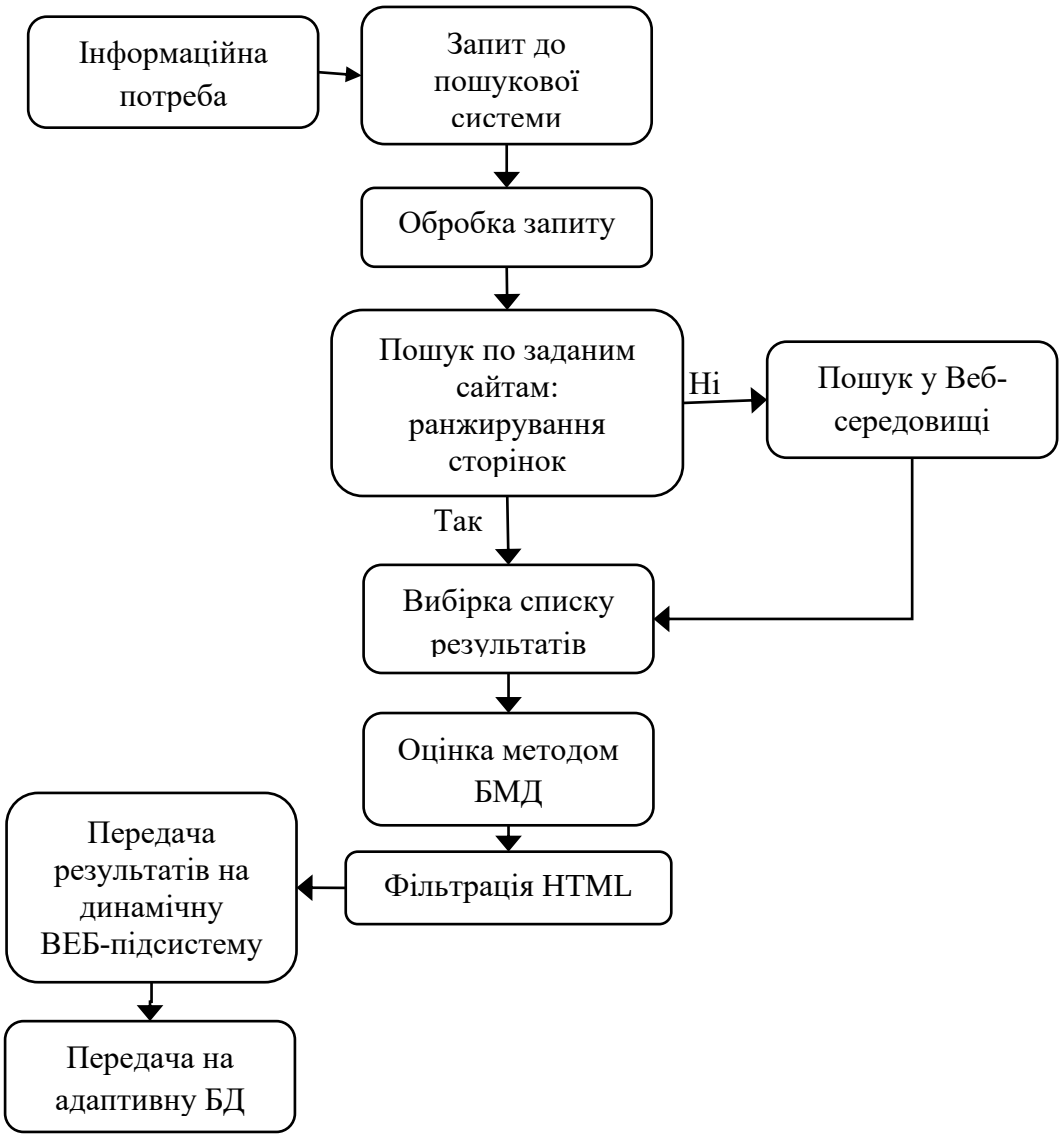

Рисунок 4.1 – Загальний алгоритм роботи пошукової системи.

Розглянемо призначення основних блоків роботи алгоритму.

Інформаційна потреба.

Іноді виникає потреба, коли ціль, яка стоїть перед користувачем у процесі його професійної діяльності не може бути виконана без залучення допоміжної інформації. Інформаційна потреба згенерована адаптивною базою даних (АБД) несе в собі команди, які потрібно виконати, а також відповідні дані з критеріями пошуку в Інтернеті у змінній question. Після чого проводиться відділення команди від змінної і запис її у змінну action, для того, щоб система була впевнена від кого отримує запит.

На основі отриманого масиву, у якому містяться показники лічильників, буде проводитись пошук, тому цей масив, названий множиною X, записується у змінну data. Ця змінна повертається до пошукової системи.

Рисунок 4.2 – Блок-схема роботи блоку інформаційної потреби.

Запит до пошукової системи.

Пошукова система приймає змінну data з даними інформації системи. Дані розбиваються на масив з показниками датчиків. Після цього вони проходять перевірку на валідність. Якщо процес валідності завершується успішно, то дані записуються знову у змінну data і передаються до блоку «Обробка запиту».

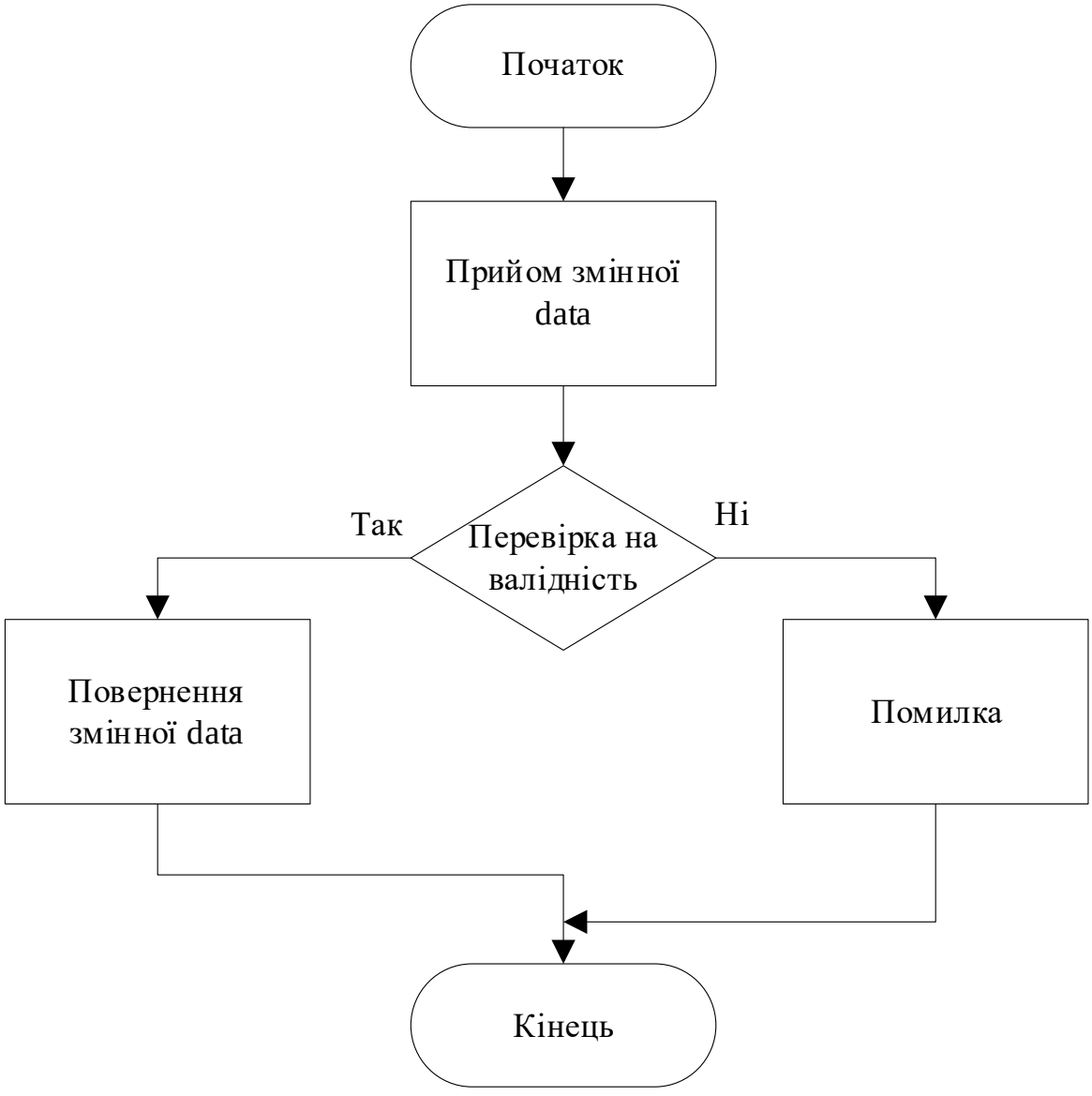

Рисунок 4.3 – Блок-схема роботи блоку запиту пошукової системи

Обробка запиту.

Перед початком пошуку, потрібно провести перевірку запиту на помилки, інакше результат буде не вірним, або взагалі не буде нічого знайдено. Для цієї задачі у даному блоці використовується умовний оператор, який перевіряє коректність до регулярного виразу. Структура виразу представлена у вигляді:

([A-Za-z_]+):([0-9.]+):([a-z]+);([A-Za-z_]+):([0-9.]+):([a-z]+);([A-Za-z_]+):([0-9.]+):([a-z]+);([A-Za-z_]+):([0-9.]+):([a-z]+)

За допомогою цього виразу запобігається можливість помилкового запиту.

У разі невідповідності - повертає помилку.

Пошук по заданим сайтам.

Після перевірки запиту, щоб провести пошук по заданим сайтам, попередньо потрібно їх завантажити на сервер і провести певну обробку, для більш зручнішого пошуку. Цей метод називається ранжируванням.

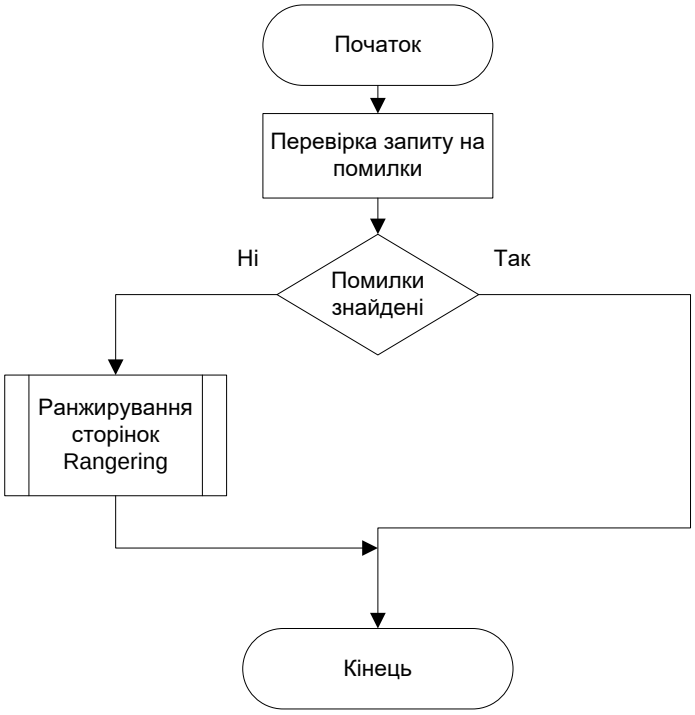

Рисунок 4.4 – Блок-схема роботи блоку пошуку по заданим сайтам

Ранжирування виконується методом перебору всього наявного текстового контенту на сайті і обробка згідно рангу важливості як показано на рисунках 4.5-4.8. Після завершення операції результат передається до наступного блоку.

Рисунок 3.5 – Загальна схема ранжирування веб-сайту

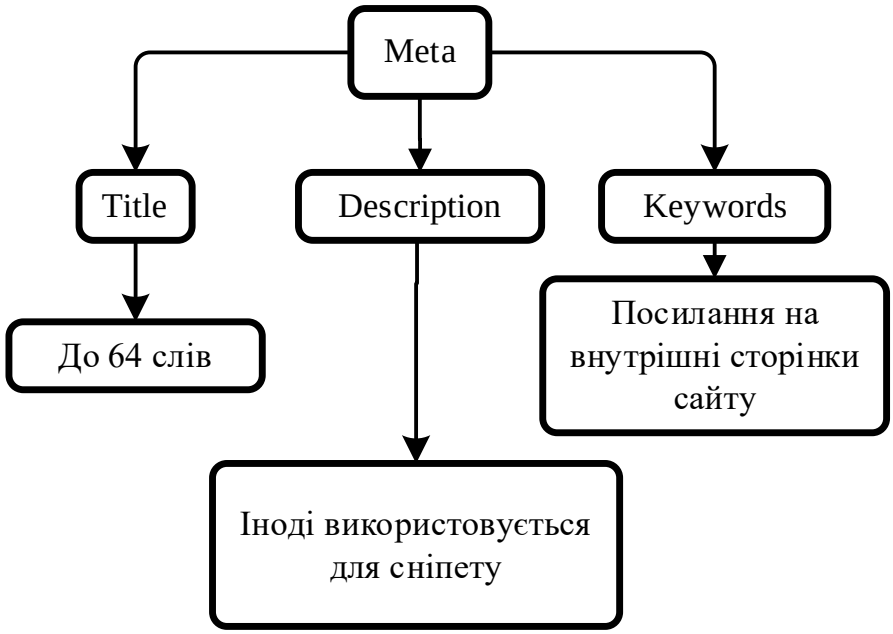

Рисунок 4.5 – Схема ранжирування веб-сайту по Meta-тегам

Рисунок 4.6 – Схема ранжирування веб-сайту через зв’язок посилань

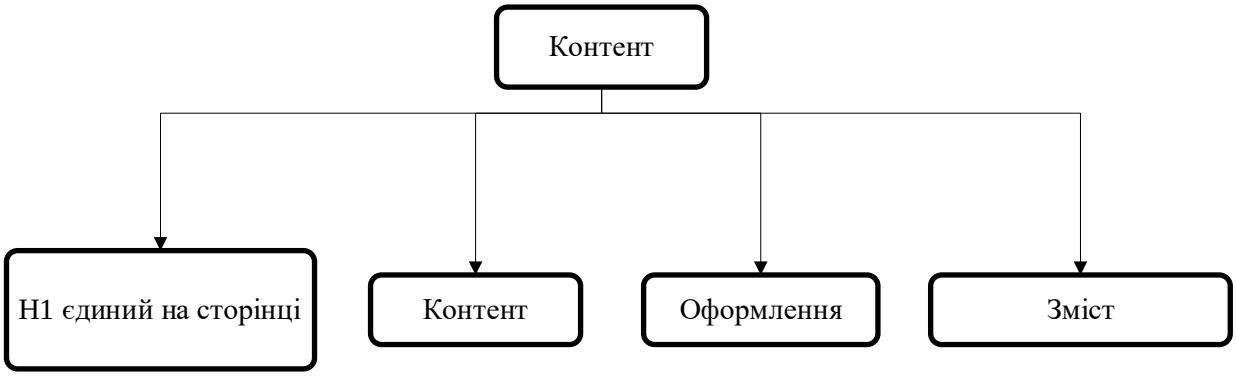

Рисунок 4.7 – Схема ранжирування веб-сайту через зв’язок посилань

Вибірка списку результатів.

Пошукова система після проведеного пошуку повинна відсортувати результат по кількості збігів. Приймає пустий масив rangering_res. Потім після створення змінної max відкривається цикл while, який пряцює з масивом rangering_res. За допомогою цього циклу перебираються сторінки результату пошуку і сортуються по мірі спадання збігів на сайті, записуючи їх по черзі в комірки массиву.

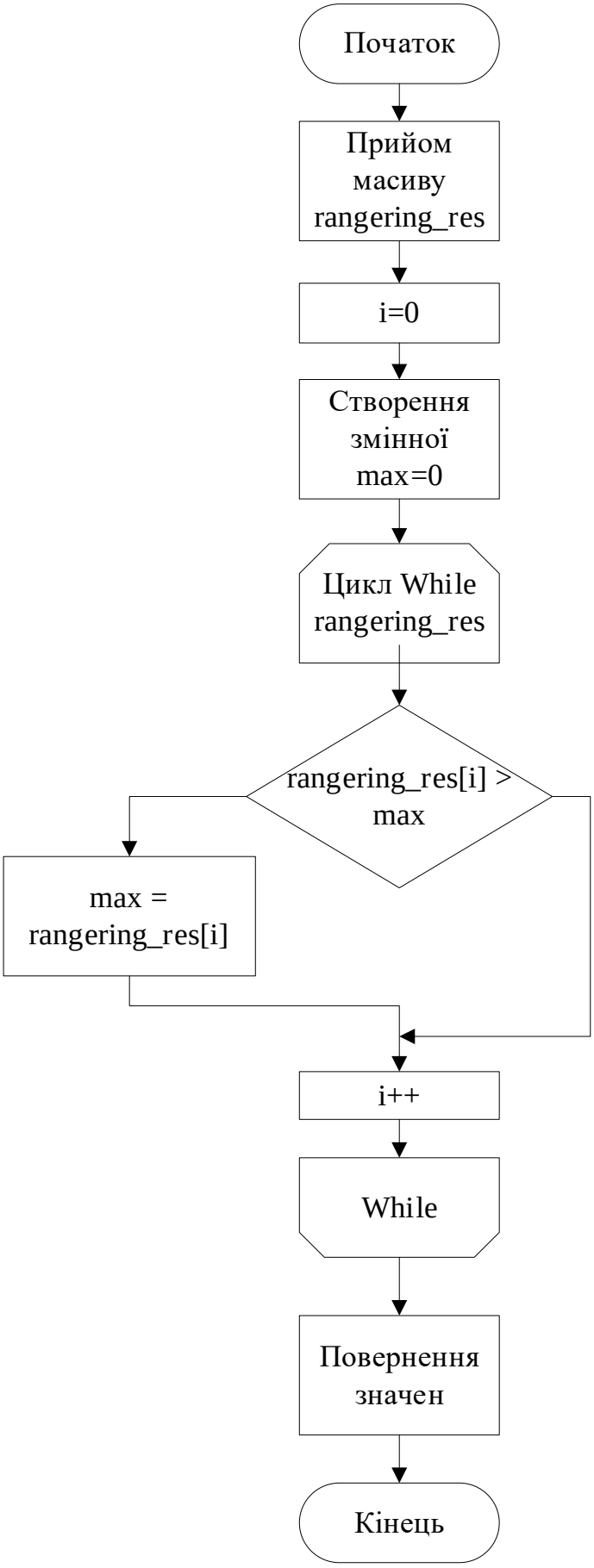

Рисунок 4.8 – Сортування сайту по кількості збігів

Оцінка релевантності запиту методом Байєсовської мережі довіри.

Об’являється порожній масив array та індекс і. За допомогою аналізу релевантності сайту на основі байесівської мережі визначаються і задаються ймовірності для кожного фактору.

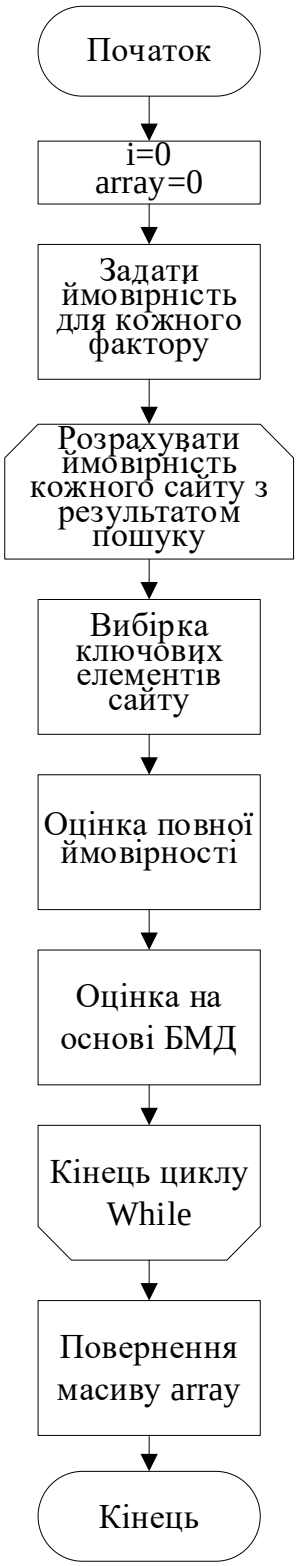

Рисунок 4.9 – Оцінка результатів пошуку на основі БМД

Алгоритм фільтрації HTML

Якщо вхідний HTML не фільтрувати або фільтрувати недостатньо якісно, виникає велика загроза безпеки сайту - з'являється можливість впровадження XSS ін'єкції. Не можна робити занадто глибокий аналіз, інакше на великих обсягах даних можлива відмова сервера. Алгоритм побудований таким чином:

- знешкоджується XCC ін’єкція;

- видаляються недопустимі теги і теги, які не відносяться до стандарту W3;

- видалення недопустимого тексту між тегами

- закриття незакритих тегів.

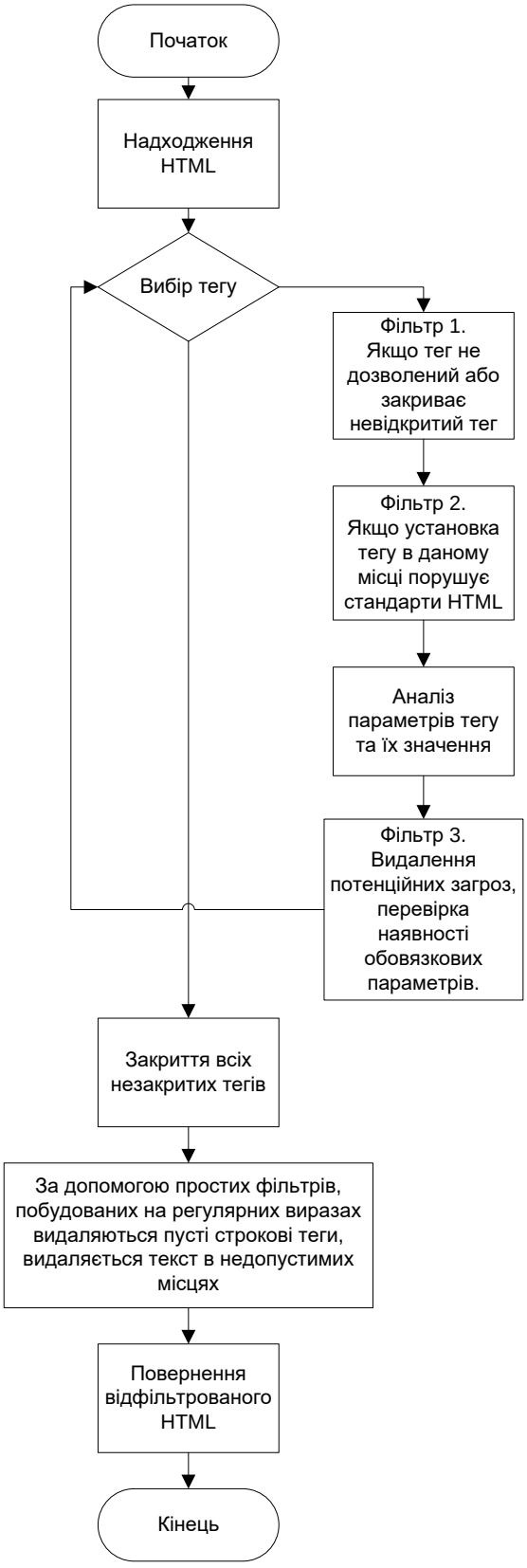

Рисунок 4.10 – Внесення результату у html формат

4.2 Вибір та обгрунтування програмного забезпечення системи.

Розглянемо основні особливості застосування операційної системи Linux. Данна ОС є з відкритими джерельними кодами та з вільним програмним забезпеченням. Її джерельні коди доступні усім для використання, модифікації та розповсюдження абсолютно безкоштовно. Linux створений Лінусом Торвальдсом, студентом з Університету Гельсінкі. Проте перша версія вийшла не дуже вдалою, потребувала присутності на комп'ютері іншої операційної системи для можливості загрузки Linux.

Назва Linux походить від прізвища розробника ядра та додавання літери «X», як це було популярно в UNIX-системах. Ядро Linux є успішним прикладом співпраці різних за розміром і вироблюваною продукцією компаній і індивідуальних розробників. Кількість цих розробників і розподіл їх внеску в розробку може служити, в деякому розумінні, гарантією стабільності і незалежності розробки. Це наочно демонструє переваги як відкритої моделі розробки в цілому, так і гарантій, що надаються розробникам і кінцевим користувачам самою концепцією вільного програмного забезпечення.

Переваги відкритого джерельного коду Linux очевидні: якщо не влаштовує щось, це легко можна змінити за умови наявності певних навичок у програмуванні. До того ж це досить надійна операційна система, для Linux майже немає вірусів.

Програмне забезпечення для Linux працює стабільно. Навіть коли користувач уже звик до Microsoft Windows, є багато аналогів звичних програм. Останнім часом дуже зросла популярність Linux, тому більшість компаній-розробників програмного забезпечення виготовляють його і для Linux теж. Великий вклад в розвиток Linux зробила компанія Adobe. Звісно доведеться звикати до програм-аналогів, але їх зручність значно прискорює цей процес. Також майже всі Windows-програми запускаються на Linux з допомогою програми Wine, що також є суттєвим плюсом Linux. Wine – це вільне програмне забезпечення, що дозволяє користувачам UNIXоподібних систем виконувати 16-, 32- та 64- бітні додатки Windows. Wine не є емулятором комп'ютера, це альтернативна реалізація Windows API. Wine приймає

системні виклики Windows-програм до бібліотек операційної системи та заміняє їх на свої. Тому емуляції процесора не відбувається, і програми працюють майже так само швидко, як і в Windows. Для роботи Wine не вимагає встановленого Windows, хоча може користуватися і її бібліотеками. Проте це не стабільна програма, і не можна говорити, що абсолютно всі програми будуть працювати, проте більшість із них стабільно працюють, що значно полегшує користувачам роботу з Linux.

Операційну систему Ubuntu прийнято вважати най лояльнішою по відношенню до користувачів. Саме принцип лояльності, справедливості та рівноправності брали за основу її творці з компанії Canonical, що є також і головним спонсором власних програмних розробок. У чому ж полягають основні принципи дружнього ставлення до користувача? Перш за все, це програмне забезпечення розповсюджується безкоштовно. Встановлювальні диски можна замовити поштою, при цьому всі витрати, пов'язані з пересилкою, обробкою замовлення і навіть митні збори компанія розробник бере на себе. Версії оновлюються з частотою в 6 місяців. Крім того, Ubuntu універсальна - ця операційна система без проблем працює як на домашньому комп'ютері, так і на серйозній робочій станції або сервері. Ubuntu можна також встановити і на ноутбук. Фактично, встановлюючи Ubuntu, користувач має під рукою всі необхідні інструменти для роботи на своєму комп'ютері. Стандартний набір додатків включає в себе пакет офісних програм, що дають можливість працювати з текстами, електронними таблицями, а також презентаціями. Програми дозволяють переглядати і редагувати існуючі файли або створювати нові.

Взагалі, перехід з Windows на Linux найпростіше здійснюється саме за допомогою Ubuntu. Ті ж офісні програми дозволяють працювати з форматами даних, сумісними з Microsoft Office. Нескладно переконатися в тому, що переваги ОС Ubuntu по достоїнству оцінені в усьому світі.

Привілеї операційної системи Linux над Windows [20]:

Віддалене управління.Linux має дуже розвинені засоби віддаленого управління. Причому керувати машиною під управлінням Linux можна з будь-якої іншої системи, де є програма емулятор терміналу.

Якщо машина підключена до Інтернету, то керувати нею можна практично з будь-якої іншої машини, також підключеної до Інтернету, швидке підключення не потрібно. Віддалене управління робочими станціями скорочує витрати на адміністрування мережі, оскільки системному адміністратору не потрібно навіть вставати зі стільця для того, щоб, наприклад, поставити будь-яке програмне забезпечення на всі робочі станції з Linux. Графічна середа підтримує відображення графіки на іншій машині, і навіть запуск різних додатків з різних систем з відображенням їх на одному екрані. При цьому додатки зберігають можливість взаємодіяти між собою (наприклад, мають загальний буфер обміну).

Доступ.

Найбільш часто використовуваним типом доступу до сервера є FTP, який забезпечують обидві системи - Windows і Linux. У даному аспекті різниця між операційними системами полягає в тому, що Linux підтримує SSH і telnet, в той час як Windows - тільки telnet.

Типи файлів.

Як Windows, так і Linux підтримує такі стандарти файлів - HTML, Cold Fusion і JavaScript. Спочатку на Linux-сервері неможливо було використовувати FrontPage розширення, але зараз даний недолік був виправлений, тому стало можливим завантажувати тип файлів FrontPage і на Linux-серверах. Що стосується CGI і Perl платформ, то вони частіше підтримуються операційною системою Linux. Якщо потрібно використовувати один з даних типів файлів, потрібно переконатися, що система підтримує дані файли. Зазвичай PHP підтримує Linux, в той час як ASP - Windows.

Бази даних [21].

Найчастіше, різницю в операційних системах можна знайти саме в питанні, пов'язаним з базами даних. Бази даних Access підтримує тільки

Windows. Бази даних MySQL знаходять найбільш часте застосування на Linux-серверах, хоча вони також підтримуються і Windows-серверами. Багато користувачів, використовуючи access бази даних, віддадуть перевагу завантажувати windows-сервер і навпаки.

4.2.1 Вибір мови програмування.

PHP (від анг. PHP: Hypertext Preprocessor - «PHP: препроцесор гіпертексту») - скриптова мова програмування загального призначення, що застосовується для розробки Web-додатків [22].

В даний час підтримується переважною більшістю хостинг-провайдерів і є одним з лідерів серед мов програмування, що застосовуються для створення динамічних Web-сайтів.

Мова і його інтерпретатор розробляються групою ентузіастів у рамках проекту з відкритим кодом. Проект не є вільним і поширюється під власною ліцензією.

У області програмування для Мережі, PHP - один з найпопулярніших скриптових мов (разом з JSP, Perl і мовами, використовуваними в ASP.NET) завдяки своїй простоті, швидкості виконання, багатій функціональності, платформ і розповсюдженню початкових кодів на основі ліцензії PHP.

Популярність у галузі побудови Web-сайтів, визначається наявністю великого набору вбудованих засобів для розробки Web-додатків. Основні з них:

Автоматичне витяг POST і GET-параметрів, а також змінних оточення Web-сервера в зумовлені масиви;

Файлові функції успішно обробляють як локальні, так і віддалені файли;

Автоматична відправка HTTP-заголовків;

Робота з cookies і сесіями;

Обробка файлів, що завантажуються на сервер;

Робота з HTTP заголовками і HTTP авторизацією;

Робота з XForms;

Робота з віддаленими файлами і сокетами.

В даний час PHP використовується сотнями тисяч розробників. Згідно з рейтингом Tiobe, що базується на даних пошукових систем, в жовтні 2009 року PHP знаходиться на 3 місці серед мов програмування (поступаючись Java і C), піднявшись за рік на дві позиції.

4.2.2 Вибір системи керування

На сьогодні СУБД MySQL забезпечується підтримкою великої кількості типів таблиць де користувачі можуть вибрати як таблиці типу MyISAM, що підтримують повнотекстовий пошук, так і таблиці InnoDB, що підтримують транзакції на рівні окремих записів.

Важливо відзначити, що на офіційному сайті СУБД для вільного завантаження надаються не тільки початкові коди, але і відкомпільовані і оптимізовані під конкретні операційні системи готові виконуючі модулі СУБД MySQL.

Вибір модуля для пошукової системи.

Так як розробка пошукової системи з нуля – це дуже трудомістка робота, була вибрана в якості модуля для пошукової системи Sphinx. Sphinx це безкоштовна пошукова система з відкритим вихідним кодом, призначена для виключно швидкого пошуку тексту[23]. Наприклад, в реальній базі даних, що складається приблизно з 300 000 строк і п'яти індексованих стовпців, де кожен стовпець містить близько 15 слів, Sphinx може видати результат з пошуку "будь-яке з цих слів" за одну соту частки секунди (на сервері з процесором AMD Opteron з частотою 2 ГГц і 1 Гб пам'яті, під керуванням Debian Linux).

У Sphinx реалізовано безліч функцій, в тому числі [24]:

- індексувати будь-які дані, представимо у вигляді рядків.
- індексувати одні й ті ж дані різними способами. Створивши кілька індексів, кожен з яких налаштований на вирішення певної задачі, ви можете вибрати найбільш відповідний для оптимізації результатів пошуку.
- пов'язувати атрибути з кожним елементом індексованих даних. Згодом можна використовувати один або кілька атрибутів для фільтрації результатів пошуку.

- підтримує морфологічні варіації слів, тому пошук за словом "cats" також видасть результат по первинній словоформе "cat".

- індекс Sphinx можна розподілити за кількома машинам, забезпечивши відмовостійку роботу.

- створювати індекси префіксів слів довільної довжини та індекси Інфікси різної довжини. Наприклад, номер прийняття рішення може складатися з 10 символів. Індекс префікса буде містити всі можливі підрядка, що починаються з початку рядка. Індекс інфіксу буде містити підрядки, що містяться в будь-якому місці рядка.

- Sphinx можна запустити в якості системи зберігання в рамках MySQL V5, що виключає необхідність запуску ще одного домену, який часто розглядається як додаткова точка збою.

Sphinx складається з трьох компонентів: генератор індексу, пошукова система і пошукова утиліта, що працює в командному рядку:

- Генератор індексу називається индексатором (indexer). Він виконує запити до бази даних, індексує кожну колонку в кожному рядку результату і прив'язує кожну запис індексу до первинного ключу рядка.

- Пошукова система являє собою домен, який називається searchd. Домен отримує критерії пошуку та інші параметри, проходить по одному або декільком індексам і повертає результат. Якщо відповідність знаходиться, searchd повертає масив первинних ключів. Використовуючи ці ключі, додаток може виконати запит за відповідною базою даних і знайти повні записи, що задовольняють критеріям пошуку. Searchd взаємодіє з додатком через сокет на порту 3312[25].

Зручна утиліта search дозволяє виконувати пошук з командного рядка без написання коду. Якщо searchd повертає результат, пошукова система формує запит до бази даних і виводить рядки, що містяться в результуючому множині. Утиліта search корисна для налагодження конфігурації Sphinx і виконання імпровізованих пошукових запитів.

4.3 Розробка коду програмного забезпечення.

```
<?php
  include('sphinx-0.9.7/api/sphinxapi.php');


  $cl = new SphinxClient();
  $cl->SetServer( "localhost", 3312 );
  $cl->SetMatchMode( SPH_MATCH_ANY  );
  $cl->SetFilter( 'model', array( 3 ) );


  $result = $cl->Query( 'cylinder', 'catalog' );


  if ( $result === false ) {
     echo "Query failed: " . $cl->GetLastError() . ".\n";
  }
  else {
     if ( $cl->GetLastWarning() ) {
        echo "WARNING: " . $cl->GetLastWarning() . "
";
     }
     if ( ! empty($result["matches"]) ) {
        foreach ( $result["matches"] as $doc => $docinfo ) {
       echo "$doc\n";          }


        print_r( $result );
     }
  }
```

```
 exit;
?>
char strstr(char big, char little) {
 char x, y, z;
 for (x = big; x;  x++) {
 for (y = little, z = x; *y; ++y, ++z) {
 if (y != z)    // попарно порівняти символи Y і Z
 break;
 }
 if (!y)
 return x;
 }
 return 0;
}
```

Висновки до четвертого розділу. В даному розділі проведена розробка програмно – алгоритмічного забезпечення пошукової системи. Розроблена загальна структура пошукової системи та алгоритми її роботи.

Наведено призначення основних блоків роботи алгоритму: інформаційної потреби, запиту пошукової системи, пошуку по заданим сайтам, ранжирування веб-сайту, ранжирування веб-сайту по Meta-тегам, сортування сайту по кількості збігів. Наведений алгоритм оцінки релевантності запиту методом Байєсовської мережі довіри.

Проведено вибір та обгрунтування програмного забезпечення системи, та розробку коду програмного забезпечення.

ВИСНОВОК

У даній роботі розроблена система пошуку та оцінки релевантних документів при побудові предметно–орієнтованої експертної системи з використанням WEB технологій для підвищення якості прийняття рішень.

Для реалізації даного завдання в роботі проведений аналіз сучасних підходів до побудови предметно–орієнтованих експертних систем та визначені основні фактори впливу на процес пошуку інформації.

Досліджено та проведено оцінку основних факторів впливу по ступеню важливості з використанням знань експертів-фахівців, за результатами яких розроблена математична модель процесу пошуку інформації та проведена оцінка міри релевантності документу запиту користувача з застосуванням ймовірнісного методу Байесовської мережі довіри.

В роботі також розроблені програмно-алгоритмічні засоби для реалізації системи пошуку та оцінки релевантних документів при побудові предметно–орієнтованої експертної системи з використанням WEB - технологій.

Результати роботи можуть бути використані в управлінських, бізнесових, наукових сферах та інших предметно – орієнтованих експертних системах для підвищення якості прийняття рішень.

СПИСОК ЛІТЕРАТУРИ

1. Павленко Ю.С. Пошукова оптимізація, технології та сервіси веб-аналітики : конспект лекцій [Електронний ресурс] / Ю.С. Павленко; ВНУ імені Лесі Українки. – Електронні текстові дані (1 файл: 968 КБ). – Луцьк: ВНУ імені Лесі Українки, 2022. – 51 с.

1. Павленко Ю.С. Основи пошукової оптимізації і технології та сервіси вебаналітики: електронний курс навчальної дисципліни. Луцьк : ВНУ ім. Лесі Українки, 2022. URL: https://moodle-cs.vnu.edu.ua/course/view.php?id=70.

2. Розкрутка Сайту — Безкоштовно на RozkrutkaSite. URL: https://rozkrutka.site/.

3. Безкоштовний курс SEO [2022] в Onpage School - Лучший бесплатный сео курс в Украине: Киев, Днепр, Одесса, Харьков, Львов - Onpage School. URL: https://onpage.school/kurs-seo-free/.

4. Гнилякевич-Проць, І., & Зінькова, С. (2022). Оцінювання ефективності оптимізації та просування вебсайту за трафіковими та конверсійними технологіями. *Підприємництво та інновації*, (24), 77-82. https://doi.org/10.32782/2415-3583/24.12

5. Іванечко Н.Р., Окрепкий Р.Б., Павелко В.І. SEO оптимізація: семантичне ядро. *Проблеми системного підходу в економіці*. Випуск №1(87), 2021. С. 109-114. DOI: https://doi.org/10.32782/2520-2200/2022-1-16.

6. Дикань В.А. Застосування наскрізної аналітики для оцінки ефективності стратегії просування в інтернеті. *B2B маркетинг* : праці XII Всеукраїнської конф., м. Київ, 2018. С. 88–89. URL: http://marketing.kpi.ua/files/b2b/%D0%97%D0%B1%D1%96%D1%80%D0%BD%D0%B8%D0%BA_B2B-2018.pdf#page=88

7. SEO for Beginners: An Introduction to SEO Basics. SEJ. URL: https://www.searchenginejournal.com/seo-guide/.

8. SEO Basics: Beginner’s Guide to SEO Success. *ahrefs blog*. URL: https://ahrefs.com/blog/seo-basics/.

9. http://emerecu.ukma.kiev.ua/books/InfSys/1/1-1-3.htm

10. Steele R. Techniques for Specialized Search Engines // Proceedings of Internet Computing '01.- Las Vegas, NV, USA, 2001.

11. Friedman N., Goldszmidt M. Building classifiers using Bayesian networks. In Proceedings of the National Conference on Artificial Intelligence ’96.- AAAI Press, Menlo Park, CA, USA, 1996. - P. 1277 -1284.

12. Kaerulff U., Jensen F.V. Bayesian networks / Tech. Rep. Department of Computer Science, Aalborg University, Denmark, 1996. - 5 p.

13. Chau M. Spidering and Filtering Web Pages for Vertical Search Engines / Proceedings of The Americas Conference on Information Systems.- AMCIS 2002 Doctoral Consortium, Dallas, TX, USA, 2002.

14. Офіційний сайт MySQL - http://mysql.com

15. Офіційний сайт PostgreSQL - http://www.postgresql.org/

16. Офіційний сайт PHP – php.net

17. Офіційний сайт ASP.NET – http://www.aspnetcms.info/

18. Експертні системи: навчальний посібник / Л.Нікітіна – Харків: НТУ «ХПІ», 2023. – 210 с

19. Introduction to CLIPS. [Електронний ресурс]. – Режим доступу: http://ir.nuk.edu.tw: 8080/ir/bitstream/310360000Q/11342/2/CLIPS_IntroCLIPS. Pdf

20. Introduction to Expert Systems. [Електронний ресурс]. – Режим доступу: https://nou.edu.ng/coursewarecontent/CIT474%20Expert%20systems.pdf

21. Introduction to Expert Systems. [Електронний ресурс]. – Режим доступу: http://www.sci.brooklyn.cuny.edu/~dzhu/cis718/preview01.pdf

22. .Nilsson N.J. The quest for artificial intelligence. // [Електронний ресурс]:Stanford University. – Режим доступу: http://ai.stanford.edu/~nilsson/QAI/qai.pdf

23. Poole D. Artificial Intelligence // [Електронний ресурс]: D. Poole, A.Mackworth – Режим доступу: http://artint.info/html/ArtInt.html.

24. Ray Barrera, Aung Sithu Kyaw, and Thet Naing Swe titled Unity 2017 Game AI Programming – Third Edition.

25. Smart Computing and Self-Adaptive Systems. Edited BySimar Preet Singh, Arun Solanki, Anju Sharma, Zdzislaw Polkowski, Rajesh Kumar CRC Press, 2022 – 288pp.