МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ДЕРЖАВНИЙ ЗАКЛАД

„ЛУГАНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ІМЕНІ ТАРАСА ШЕВЧЕНКА”

Навчально-науковий інститут фізики, математики та інформаційних технологій

Кафедра інформаційних технологій та систем

Гращенков Данило Денісович

**ДОСЛІДЖЕННЯ ТЕХНОЛОГІЙ РОЗРОБКИ ЕЛЕКТРОННИХ СЕРВІСІВ ПОШУКУ**

**кваліфікаційна робота**

**здобувача вищої освіти першого (бакалаврського)** **рівня**

**освітньої програми «Інженерія програмного забезпечення»**

**за спеціальністю 121 Інженерія програмного забезпечення**

Особистий підпис \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Данило ГРАЩЕНКОВ

Науковий керівник \_\_\_\_\_\_\_\_\_\_\_\_\_ Микола СЕМЕНОВ,

кандидат технічних наук, доцент

кафедри інформаційних технологій

та систем

Завідувач кафедри \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Микола СЕМЕНОВ,

кандидат педагогічних наук, доцент

кафедри інформаційних технологій

та систем

Полтава – 2023

Міністерство освіти і науки України

Державний заклад „Луганський національний університет   
імені Тараса Шевченка”

|  |  |
| --- | --- |
| Факультет (інститут) | Навчально-науковий інститут фізики, математики та інформаційних технологій |
| Кафедра | Інформаційних технологій та систем |
| Рівень освіти | перший (бакалаврський) |
| Спеціальність | 121 «Інженерія програмного забезпечення» |
| (код, назва) |

|  |  |  |  |
| --- | --- | --- | --- |
|  | ЗАТВЕРДЖУЮ  Завідувач кафедри ІТС | | |
|  |  | Микола СЕМЕНОВ |
| (підпис) (ім'я, прізвище) | | |
| “\_\_\_”\_\_\_\_\_\_\_\_\_\_\_\_\_2023 р. | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ЗАВДАННЯ**  **НА КВАЛІФІКАЦІЙНУ РОБОТУ** | | | | | |
| Гращенкова Данила Денісовича | | | | | | |
| *(*прізвище, ім’я, по батькові *)* | | | | | | |
| **1.** Тема проекту (роботи) | | Дослідження технології розробки електронних | | | | |
| сервісів пошуку | | | | | | |
|  | | | | | | |
| Керівник кваліфікаційної роботи | | | | Семенов М.А. | | |
|  | | | | (прізвище, ім’я, по батькові, науковий ступінь, вчене звання) | | |
| затверджена наказом по університету | | | | | Від“\_\_” \_\_\_\_\_\_\_ 2023 року№\_ | |
| **2.** Строк подання студентом проекту (роботи) | | | | |  | |
| **3.** Вихідні дані до роботи(проекту) | | | у результаті виконання роботи | | | |
| повинно бути розроблено метод ранжування пошукових запитів на основі Okapi BM25 | | | | | | |
|  | | | | | | |
|  | | | | | | |
|  | | | | | | |
|  | | | | | | |
| (визначаються кількісні або (та) якісні показники, яким повинен відповідати об’єкт розробки) | | | | | | |
| **4.** Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно | | | | | | |
| розробити) | АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ. | | | | | |
| ПОРІВНЯЛЬНУ ХАРАКТЕРИСТИКА | | | | | | |
| РЕКОМЕНДАЦІЇ ЩОДО ОПТИМІЗАЦІЇ НАЯВНИХ ТЕХНОЛОГІЙ | | | | | | |
| РОЗРОБКИ ЕЛЕТКРОННИХ СЕРВІСІВ ПОШУКУ. | | | | | | |
|  | | | | | | |
| (визначаються назви розділів або (та) перелік питань, які повинні увійти до тексту ПЗ) | | | | | | |
| **5.** Перелік графічного матеріалу (з точним зазначенням обов’язкових креслень) | | | | | | |
|  | | | | | | |
|  | | | | | | |

**6.** Консультанти розділів проекту (роботи)

|  |  |  |  |
| --- | --- | --- | --- |
| Розділ | Прізвище, ініціали та посада  Консультанта | Підпис, дата | |
| завдання видав | завдання прийняв |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**7.** Дата видачі завдання „ ” 2023 р.

**КАЛЕНДАРНИЙ ПЛАН**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| №  з/п | Назва етапів дипломного  проекту (роботи) | | | | Строк виконання етапів проекту (роботи) | Примітка | |
|  | Вибір теми роботи, вивчення наукової літератури, затвердження теми та керівника. | | | | До 15 жовтня |  | |
|  | Аналіз літературних джерел за темою роботи. Розробка та апробація методики дослідно-експериментальної роботи.  Подання структури теоретичної частини роботи та плану експериментальних досліджень. | | | | Другий тиждень листопада  (10 листопада ) |  | |
|  | Робота над теоретичною частиною.  Подання теоретичної частини роботи для першого читання науковим керівником. | | | | До 15 грудня |  | |
|  | Усунення зауважень, урахування рекомендацій наукового керівника.  Подання теоретичної частини роботи на друге читання. | | | | До 28 січня |  | |
|  | Проведення експериментальної роботи. Поетапний аналіз та обговорення її результатів. Перевірка стану виконання роботи. | | | | Перший тиждень березня |  | |
|  | Урахування рекомендацій наукового керівника, усунення недоліків, підготовка варіанта роботи до передзахисту.  Розробка презентації. | | | | До 31 березня |  | |
|  | Попередній захист роботи на кафедрі | | | | квітень |  | |
|  | Доопрацювання роботи з урахуванням рекомендацій після передзахисту. Подання роботи науковому керівникові та рецензентові на підготовку відгуку та рецензії | | | | За 10 днів до державної атестації |  | |
|  | Подання на кафедру остаточного варіанта роботи, переплетеного та підписаного автором, науковим керівником і рецензентом. | | | | За 5 днів до державної атестації |  | |
|  | |  |  |  | | |
| **Студент** | |  |  | Данило ГРАЩЕНКОВ | | |
| підпис |  |  | | |
| **Керівник проекту (роботи)** | |  |  | Микола Семенов | | |

підпис

**АНОТАЦІЯ**

**Гращенков Д.Д.**

**Тема:** **Дослідження технологій розробки електронних сервісів пошуку.**

**Спеціальність:** 121 "Інженерія програмного забезпечення"

**Установа:** ЛНУ імені Тараса Шевченка, 2021 р.

**Бакалаврська робота містить:** 75 с., 5 рис., 5 табл., 3 додат., 31 джерел.

**Об’єкт дослідження:** технології розробки системи цифрового пошуку.

**Предмет дослідження:**моделі, алгоритми, інформаційно-пошукові мови та методи ранжування у контексті розробки системи цифрового пошуку.

**Мета роботи** – аналіз технологій розробки електронних сервісів пошуку.

**Результати роботи.** В роботі проведено порівняльний аналіз наявних методів та алгоритмів ранжування пошукових результатів, а також виконано проектування і розробка алгоритму оптимізації на основі OkamiBM25.

**Ключові слова.** ПОШУКОВИ СИСТЕМИ, МЕТОДИ РАНЖУВАННЯ, ІНФОРМАЦІЙНО-ПОШУКОВІ МОВИ, ЦИФРОВІ СЕРВІСИ, OKAMIBM25.

**ABSTRACT**

**Pantsyr S.R.**

**Theme:** **Development of a two-dimensional game for ANDROID using Microsoft Visual Studio.**

**Speciality:** 121 "Software Engineering"

**Institution:** Luhansk Taras Shevchenko National University (LTSNU), 2021.

**Diploma work contains:** 56 pages, 31 Fig., 1 Table, 2 adj., 37 source.

**A research object is** mobile games, which have taken a leading position in the entertainment market, leaving behind personal computers and consoles.

**The article of research is** creation taking into account constant requirements of consumers to innovations, the realistic game developed for the Android operating system.

**An aim of work is** development of a two-dimensional game for Android using Microsoft Visual Studio.

**Job performances.** The analysis of statistical data related to the development of mobile games, as well as the design and development of a two-dimensional game on the Android mobile platform.

**Grashchenkov D.D.**

**Theme: Research of technologies for the development of electronic search services.**

**Specialty:** 121 "Software Engineering"

**Institution:** Luhansk Taras Shevchenko National University, 2021.

**Diploma work contains:** 75 p., 5 figs., 5 tables, 3 appendices, 31 sources.

**A research object is** technologies for developing a digital search system.

**The article of research is** models, algorithms, information retrieval languages and ranking methods in the context of developing a digital search system.

**An aim of work is** to analyze the technologies for the development of electronic search services.

**Job performances.** A comparative analysis of existing methods and algorithms for ranking search results is carried out, as well as the design and development of an optimization algorithm based on OkamiBM25.

**Keywords.** SEARCH ENGINES, RANKING METHODS, INFORMATION AND SEARCH LANGUAGES, DIGIМІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

# ВСТУП

**Актуальність роботи**

З поширенням цифрового контенту та зростанням довіри до онлайнової інформації цифрові пошукові сервіси стали важливими інструментами для пошуку інформації. Ця наукова робота присвячена дослідженню технологій, що використовуються при розробці електронних пошукових сервісів. Вивчаючи різні технологічні підходи, алгоритми та архітектури, що застосовуються при розробці та впровадженні цих сервісів, це дослідження має на меті забезпечити комплексне розуміння технологічного ландшафту, що оточує електронні пошукові сервіси.

У дослідженні розглядається низка технологій, що використовуються при розробці електронних пошукових сервісів, включаючи методи пошуку інформації, методи індексування і алгоритми обробки запиті. Аналізуючи та порівнюючи ці технології, дослідження має на меті визначити їхні сильні та слабкі сторони, а також придатність для різних типів пошукових сервісів та потреб користувачів.

Крім того, дослідження вивчає вплив нових технологій, таких як машинне навчання, обробка природної мови та інтелектуальний аналіз даних, на розвиток електронних пошукових сервісів. Досліджується, як ці технології розширюють можливості пошуку, підвищують релевантність результатів пошуку та уможливлюють персоналізований пошук.

**Мета роботи**

Метою бакалаврської роботи є аналіз технологій розробки електронних сервісів пошуку.

Досягнення мети включає розв’язання таких **завдань**:

1) висвітлення поняття та класифікації технологій розробки електронних сервісів пошуку;

2) визначення сучасних обмежень та проблеми технологій розробки цифрових сервісів пошуку.

3) аналіз основних моделей та алгоритмів пошуку та інформаційно-пошукових мов;

4) наведено порівняльну характеристику методів ранжування пошукових систем№

5) розроблено рекомендації щодо оптимізації та модернізації алгоритмів пошуку та методів ранжування.

**Об’єктом дослідження** є технології розробки системи цифрового пошуку.

**Предметом дослідження** є моделі, алгоритми, інформаційно-пошукові мови та методи ранжування у контексті розробки системи цифрового пошуку.

**Практичне значення отриманих результатів**

Шляхом поглибленого аналізу наявної літератури, тематичних досліджень та інтерв'ю з експертами ця робота має на меті зробити внесок в існуючий масив знань про технології розвитку електронних пошукових сервісів. Висновки та ідеї, отримані в результаті цього дослідження, можуть допомогти постачальникам пошукових послуг, розробникам і дослідникам у прийнятті обґрунтованих рішень щодо вибору, впровадження та оптимізації технологій для покращення функціональності, продуктивності та користувацького досвіду електронних пошукових сервісів.

**Структура і обсяг роботи**

Робота складається з вступу, трьох розділів, висновків списку використаних джерел, додатків. Обсяг роботи становить 74 сторінки, обсяг використаної літератури – 31 джерела. Перший розділ охоплює аналіз концепції цифрових сервісів пошуку та класифікацію технологій, що складають архітектуру подібних систем У другому розділі проводиться дослідження та порівняльна характеристика основних компонентів систем цифрового пошуку. Третій розділ охоплює рекомендації щодо подальшого дослідження та оптимізації наявних методів та технологій розробки систем цифрового пошуку. Додатки містять список запитів, який було використано з метою проведення порівняльного аналізу методів ранжування у розділі 2.3.

# РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

## Поняття цифрових сервісів пошуку

Концепція цифрових пошукових сервісів стала важливою темою у контексті інформаційного пошуку та доступу до інформації в Інтернеті. Зі стрімким розвитком технологій та широким розповсюдженням інтернету окремі особи та організації все більше покладаються на цифрові пошукові сервіси для навігації у величезній кількості доступного цифрового контенту.

Цифрові пошукові сервіси охоплюють низку технологій, алгоритмів і методологій, що використовуються для пошуку та представлення релевантної інформації користувачам на основі їхніх пошукових запитів [16]. Ці сервіси виступають посередниками між користувачами та великими цифровими сховищами, такими як веб-сайти, бази даних та онлайн-платформи, полегшуючи ефективний та оперативний доступ до потрібної інформації.

Одним із фундаментальних аспектів цифрових пошукових сервісів є процес індексування, який передбачає організацію та каталогізацію цифрового контенту. Пошукові системи використовують складні алгоритми для сканування та індексації веб-сторінок, створюючи індекс, який уможливлює швидкий і точний пошук потрібної інформації [10]. Завдяки процесу індексування пошукові сервіси створюють всеосяжну базу даних цифрового контенту, що дозволяє користувачам знаходити інформацію, вводячи ключові слова або фрази в інтерфейс пошуку.

Крім того, цифрові пошукові сервіси використовують алгоритми ранжування для визначення релевантності та важливості знайденої інформації. Пошукові системи використовують складні алгоритми ранжування, які враховують різні фактори, такі як популярність веб-сторінки, наявність релевантних ключових слів і якість контенту [13]. Враховуючи ці фактори, пошукові сервіси прагнуть представити користувачам найбільш релевантну та достовірну інформацію, покращуючи загальний досвід пошуку.

Крім того, цифрові пошукові сервіси охоплюють функції, орієнтовані на користувача, та методи персоналізації для подальшого вдосконалення результатів пошуку. Ці функції включають пошукові фільтри, розширені параметри пошуку та персоналізовані рекомендації, засновані на вподобаннях користувача та історії переглядів [7]. Підлаштовуючи результати пошуку під конкретного користувача, цифрові пошукові сервіси мають на меті забезпечити більш персоналізований та інтуїтивно зрозумілий досвід пошуку, підвищуючи задоволеність користувачів та ефективність пошуку інформації.

Цифрові пошукові сервіси широко вивчаються дослідниками та науковцями в галузі інформаційного пошуку, комп'ютерних наук та взаємодії людини з комп'ютером. Численні вітчизняні та зарубіжні дослідники вивчали і продовжують вивчати різні аспекти пошукових алгоритмів, користувацьких інтерфейсів, ранжування релевантності та метрики оцінки інформаційного пошуку.

До видатних дослідників у галузі пошуку інформації належить Рікардо Баеза-Ятес, який зробив значний внесок у веб-пошук та інтелектуальний аналіз даних [9]. Його робота про сучасний пошук інформації дала цінне розуміння проблем і досягнень у сфері цифрових пошукових сервісів. Ще одна впливова постать – це Крістофер Д. Меннінг, відомий своїми дослідженнями в галузі обробки природної мови та статистичних моделей в інформаційному пошуку [16]. Внесок Меннінга значно вплинув на розуміння і розвиток пошукових технологій.

Ще однією з важливих сфер дослідження цифрових пошукових сервісів є вивчення персоналізованого пошуку. Дослідники вивчали методи врахування вподобань користувачів, контекстної інформації та даних соціальних мереж для покращення результатів пошуку на основі індивідуальних потреб користувачів [14]. Такі підходи до персоналізованого пошуку мають на меті покращити релевантність та якість результатів пошуку, враховуючи специфічні характеристики та інтереси кожного користувача.

Крім того, академічні пошукові системи привертають значну увагу наукової спільноти з практичної точки зору – спеціалізовані пошукові сервіси орієнтовані саме на академічну літературу та наукові роботи, надають науковцям та дослідникам ефективний доступ до наукової інформації [27]. Прикладами популярних академічних пошукових систем є Google Scholar, Microsoft Academic та arXiv.

На додаток до академічних пошукових систем з'явилися й інші спеціалізовані пошукові сервіси, призначені для конкретних сфер і потреб користувачів. Наприклад, пошукові системи для пошуку зображень використовують передові методи аналізу зображень, щоб дозволити користувачам шукати і знаходити візуальний контент [28]. Відеопошукові системи, з іншого боку, зосереджені на індексації та пошуку відеоконтенту з різних онлайн-платформ. Ці спеціалізовані пошукові сервіси підкреслюють диверсифікацію та кастомізацію цифрових пошукових послуг для задоволення конкретних типів контенту.

Постійний розвиток і вдосконалення цифрових пошукових сервісів суттєво вплинули на різні аспекти сучасного суспільства – від доступу до інформації та поширення знань до електронної комерції та процесів прийняття рішень. Інтеграція штучного інтелекту, машинного навчання та методів обробки природної мови ще більше розширює можливості цифрових пошукових сервісів, роблячи їх більш інтелектуальними та адаптивними до потреб користувачів.

Крім того, концепція цифрових пошукових послуг виходить за рамки традиційних веб-пошукових систем. Воно охоплює різні спеціалізовані пошукові сервіси, орієнтовані на конкретні сфери, такі як академічні пошукові системи, пошукові системи зображень, відеопошукові системи та пошукові системи електронної комерції [18]. Кожна з цих спеціалізованих пошукових служб використовує адаптовані алгоритми і методології для задоволення конкретних потреб і характеристик відповідної галузі.

Отже, концепція цифрових пошукових сервісів відіграє вирішальну роль у забезпеченні ефективного та результативного доступу до цифрового контенту. Використовуючи індексацію, алгоритми ранжування, функції, орієнтовані на користувача, і методи персоналізації, ці сервіси дозволяють користувачам орієнтуватися у величезному цифровому ландшафті і знаходити потрібну інформацію. Оскільки технології продовжують розвиватися, цифрові пошукові сервіси, ймовірно, розвиватимуться й надалі, покращуючи пошуковий досвід і розширюючи можливості користувачів у їхньому пошуку цифрової інформації.

## Склад та архітектура електронних сервісів пошуку

Склад та архітектура електронних пошукових сервісів відіграють ключову роль у забезпеченні ефективного і точного пошуку інформації в цифровому середовищі [9]. Ці сервіси призначені для організації, обробки та представлення релевантної інформації користувачам на основі їхніх пошукових запитів. Склад та архітектура електронних пошукових сервісів охоплюють різні компоненти та функціональні можливості, які працюють разом, щоб забезпечити безперебійний та ефективний пошук.

В основі електронних пошукових сервісів лежить компонент індексування, який передбачає систематичну організацію та каталогізацію цифрового контенту. Пошукові системи використовують складні алгоритми для сканування, аналізу та індексації веб-сторінок, створюючи індекс, який уможливлює ефективний пошук інформації. Процес індексування передбачає вилучення та зберігання метаданих, таких як ключові слова, заголовки та URL-адреси, а також фактичний вміст веб-сторінок. Така індексація дозволяє пошуковим службам швидко зіставляти запити користувачів з релевантною інформацією [30].

Архітектура електронних пошукових сервісів зазвичай складається з декількох рівнів і компонентів, які співпрацюють для надання результатів пошуку. Однією з найпоширеніших архітектурних схем є трирівнева архітектура, що складається з рівня представлення, прикладного рівня та рівня даних. Рівень представлення відповідає за взаємодію з користувачем, відображаючи пошукові інтерфейси і приймаючи пошукові запити. Прикладний рівень обробляє запити користувачів, виконує пошукові алгоритми і витягує відповідну інформацію з рівня даних. Рівень даних складається з індексованого контенту та пов'язаних з ним метаданих, що зберігаються в базах даних або розподілених файлових системах [17].

На додаток до основного компонента індексування, електронні пошукові сервіси використовують розширені алгоритми ранжування для визначення релевантності та розташування результатів пошуку. Ці алгоритми оцінюють різні фактори, такі як відповідність ключових слів, популярність, авторитетність і свіжість контенту, щоб визначити рейтинг для кожного знайденого документа. Алгоритми ранжування мають на меті представити найбільш релевантну та достовірну інформацію у верхній частині результатів пошуку, підвищуючи задоволеність користувачів та полегшуючи ефективний доступ до інформації [10].

Крім того, архітектура електронних пошукових сервісів включає в себе стратегії обробки запитів і пошуку. Обробка запитів передбачає аналіз запитів користувачів, розуміння їхніх намірів і перетворення їх на виконувані пошукові операції. Пошукові сервіси використовують такі методи, як розширення запиту, зворотний зв'язок щодо релевантності та семантичний аналіз, щоб покращити розуміння запиту і підвищити точність пошуку. Стратегії пошуку визначають найефективніший спосіб вилучення релевантної інформації з проіндексованих даних на основі запиту користувача та доступних ресурсів. Ці стратегії можуть включати паралельну обробку, розподілений пошук і механізми кешування для оптимізації продуктивності пошуку [27].

Також цифрові пошукові сервіси часто включають функції, орієнтовані на користувача, та механізми персоналізації. Функції, орієнтовані на користувача, такі як розширені пошукові фільтри, підказки до запитів і виправлення орфографії, покращують досвід пошуку, надаючи користувачам додатковий контроль і підказки для уточнення пошукових запитів. Механізми персоналізації використовують профілі користувачів, історію переглядів і відгуки, щоб адаптувати результати пошуку до індивідуальних уподобань і потреб. Завдяки персоналізації пошукові сервіси прагнуть надавати більш релевантну та персоналізовану інформацію, підвищуючи рівень задоволеності та залученості користувачів [9].

Таким чином, склад і архітектура електронних пошукових сервісів продовжують розвиватися разом із появою нових технологій та очікувань користувачів. Досягнення в обробці природної мови, машинному навчанні та аналітиці даних сприяють вдосконаленню пошукових алгоритмів, користувацьких інтерфейсів і ранжування релевантності. Поява голосового пошуку, мобільного пошуку та інтелектуальних асистентів ще більше впливає на розробку та впровадження електронних пошукових сервісів, забезпечуючи більш природний та контекстно-орієнтований пошук.

Отже, склад і архітектура електронних пошукових сервісів є ключовими компонентами в розробці ефективних і дієвих інформаційно-пошукових систем. Завдяки використанню алгоритмів індексування, ранжування, обробки запитів і функцій, орієнтованих на користувача, ці сервіси дають змогу користувачам орієнтуватися у величезному цифровому просторі та отримувати зручний доступ до потрібної інформації. З розвитком технологій і зміною потреб користувачів склад і архітектура електронних пошукових сервісів продовжуватимуть адаптуватися, що сприятиме подальшому покращенню пошукового досвіду і розширенню можливостей користувачів у їхньому пошуку цифрової інформації.

## Класифікація та характеристика технологій розробки цифрових пошукових сервісів

Електронні пошукові сервіси стали незамінними інструментами в сучасному цифровому ландшафті, забезпечуючи ефективний і точний пошук інформації. Розвиток цих сервісів спирається на різноманітні технології, які сприяють їхній функціональності та ефективності.

У таблиці 1.1 наведено характеристику ключових технологій, що використовуються в цифрових пошукових сервісах. Вона висвітлює особливості, сильні сторони та обмеження кожної категорії технологій, надаючи уявлення про їхню роль та міркування щодо розробки ефективних і зручних для користувача пошукових систем.

*Таблиця 1.1*

**Характеристика технологій для електронних пошукових сервісів**

|  |  |  |  |
| --- | --- | --- | --- |
| **Технологія** | **Особливості** | **Сильні сторони** | **Обмеження** |
| Індексація | Систематична організація даних | Забезпечує ефективний пошук | Покладання на структуровані дані |
| Рейтинг та релевантність | Визначення порядку результатів пошуку | Покращує релевантність результатів | Обмежено наявними даними |
| Обробка запитів | Трансформація запитів користувачів | Підвищує точність пошуку | Залежність від якості запиту |
| Взаємодія з користувачем | Покращує взаємодію з користувачем | Кастомізація та персоналізація | Занепокоєння щодо конфіденційності |

Джерело: [9].

Технології **індексування** передбачаютьсистематичну організацію даних, що уможливлює ефективний пошук. Вони створюють індекс, який дозволяє пошуковим сервісам зіставляти запити користувачів з релевантною інформацією. Перевагами індексування є покращена ефективність пошуку та швидший пошук інформації. Однак технології індексування покладаються на структуровані дані, що може бути обмеженням при роботі з неструктурованим або семантично різноманітним контентом.

Технології **ранжування та релевантності** визначають порядок, у якому користувачі отримують результати пошуку. Вони використовують алгоритми, які враховують такі фактори, як відповідність ключових слів, популярність джерела та авторитетність. Сильні сторони цих технологій полягають у підвищенні релевантності результатів пошуку та задоволеності користувачів. Однак їхні обмеження пов'язані із залежністю від наявних даних та упередженнями, які можуть бути притаманні алгоритмам ранжування.

Технології **обробки запитів** зосереджені на перетворенні користувацьких запитів на виконувані пошукові операції. Вони включають синтаксичний аналіз, розширення, семантичний аналіз і переписування запитів. Ці технології сприяють підвищенню точності пошуку і кращому розумінню намірів користувача. Однак на їхню ефективність впливає якість користувацьких запитів, і вони можуть зіткнутися з проблемами при обробці неоднозначних або складних запитів.

Технології **взаємодії з користувачем** мають наметі покращити загальний користувацький досвід і полегшити ефективну взаємодію з пошуковими сервісами. Вони охоплюють дизайн інтерфейсу користувача, механізми підказки запитів, розширені пошукові фільтри та функції персоналізації. Технології взаємодії з користувачем пропонують кастомізацію, персоналізацію та підвищену зручність використання. Однак при впровадженні функцій персоналізації, які покладаються на дані та поведінку користувача, можуть виникати проблеми з конфіденційністю.

Хоча таблиця, наведена на початку розділу, охоплює деякі ключові технології, існує багато додаткових технології, що мають відношення до розвитку цифрових пошукових сервісів. Наведемо кілька прикладів технологій, які не були згадані в табл. 1.1:

1. **Обробка природної мови** передбачає комп'ютерне розуміння та генерування людської мови. Технології NLP дозволяють пошуковим сервісам аналізувати та інтерпретувати запити користувачів, витягувати релевантну інформацію, виконувати семантичний аналіз і підтримувати розширені мовні функції, такі як аналіз настроїв, розпізнавання об'єктів і відповіді на запитання.

2. **Методи** **машинного навчання** відіграють важливу роль в електронних пошукових сервісах. Алгоритми ML можна застосовувати для різних завдань, включаючи ранжування та релевантність, розуміння запитів, персоналізовані рекомендації та виявлення спаму. Ці алгоритми навчаються на основі шаблонів даних і поведінки користувачів, щоб підвищити точність пошуку і надавати більш персоналізовані та релевантні результати.

3. **Показники оцінювання** маютьвирішальне значення для оцінки продуктивності та ефективності електронних пошукових сервісів. Для оцінки якості та релевантності результатів пошуку зазвичай використовують такі показники, як точність, відгуки, оцінка F1, середня точність (MAP) і нормалізований дисконтований кумулятивний виграш (nDCG). Ці показники допомагають дослідникам і розробникам оцінювати і порівнювати різні пошукові алгоритми і методи.

4. Ефективні технології **зберігання та управління даними** мають важливе значення для розвитку електронних пошукових сервісів. Сюди входять такі технології, як розподілені файлові системи, системи управління базами даних, індексування даних, механізми кешування та чисельні рішення для масштабування. Ці технології забезпечують надійне і масштабоване зберігання та пошук великих обсягів даних.

5. Технології **семантичного пошуку** спрямованіна розуміння значення і контексту запитів користувачів і пошукового контенту. Використовуючи семантичний аналіз, онтології, графи знань і міри семантичної схожості, семантичний пошук підвищує точність і релевантність результатів пошуку, враховуючи семантичні зв'язки і поняття, пов'язані з термінами запиту.

6. Технології **візуалізації** інформації полегшують представлення результатів пошуку в інтуїтивно зрозумілий і візуально привабливий спосіб. Ці технології включають інтерактивні діаграми, графіки, карти та інші візуальні представлення, які дозволяють користувачам ефективніше досліджувати і розуміти результати пошуку. Методи візуалізації інформації підвищують залученість користувачів і сприяють прийняттю рішень на основі даних.

Важливо зазначити, що сфера електронних пошукових сервісів дуже широка і постійно розвивається, і існує багато інших технологій і методологій, які можуть бути актуальними в залежності від конкретного контексту і вимог пошукового сервісу.

## Сучасні обмеження та проблеми електронних сервісів пошуку

Цифрові пошукові сервіси революціонізували пошук інформації, надавши користувачам швидкий і зручний доступ до величезних обсягів цифрового контенту. Однак, незважаючи на численні переваги, ці сервіси не позбавлені обмежень і проблем. Цей розділ висвітлює деякі з поточних обмежень і проблем, пов'язаних з електронними пошуковими сервісами, проливаючи світло на сфери, які потребують подальшої уваги та досліджень.

1. Величезний обсяг доступного цифрового контенту створює значні труднощі для електронних пошукових сервісів. Користувачі часто стикаються з надмірною кількістю результатів пошуку, що ускладнює пошук найбільш релевантної та достовірної інформації [11]. **Інформаційне перевантаження** може призвести до розчарування користувачів, зниження ефективності пошуку та зменшення їхньої задоволеності.

Крім того, перевантаження інформацією може призвести до зниження рівня задоволеності користувачів. Коли користувачі не можуть знайти інформацію, яку вони шукають, швидко та ефективно, це негативно впливає на їхній загальний досвід пошуку. Незадоволеність результатами пошуку може підірвати довіру до пошукової служби і призвести до того, що користувачі шукатимуть альтернативні засоби пошуку інформації або покладатимуться на обмежену підгрупу джерел, які вони вважають достовірними. Таке зниження рівня задоволеності може в кінцевому підсумку вплинути на лояльність користувачів і їхню взаємодію з пошуковим сервісом.

Для пом'якшення наслідків інформаційного перевантаження були запропоновані різні стратегії та методи. До них відносяться:

1) уточнення запитів: надання користувачам інструментів і функцій для уточнення пошукових запитів дозволяє їм звузити сферу пошуку і отримати більш релевантні результати.

2) фільтрація та ранжування результатів: впровадження вдосконалених алгоритмів ранжування та механізмів фільтрації може допомогти визначити пріоритетність найбільш релевантних та надійних результатів пошуку, зменшуючи когнітивне навантаження на користувачів.

3) персоналізація: адаптація результатів пошуку на основі вподобань користувача, його минулої поведінки та контексту може допомогти зменшити інформаційне перевантаження, надаючи користувачам більш цілеспрямовані та персоналізовані результати.

4) візуалізація інформації: представлення результатів пошуку у візуально привабливий та інтуїтивно зрозумілий спосіб, наприклад, за допомогою інтерактивних діаграм або візуальних зведень, може покращити здатність користувачів швидко схоплювати основні висновки та орієнтуватися у великих обсягах інформації.

5) допомога користувачам: надання контекстних підказок, рекомендацій щодо запитів або інтелектуальної допомоги в процесі пошуку може допомогти користувачам уточнити свої запити і звузити пошук до більш релевантної інформації.

Застосовуючи ці стратегії та постійно вдосконалюючи пошукові алгоритми і користувацькі інтерфейси, постачальники пошукових послуг можуть пом'якшити проблеми, пов'язані з інформаційним перевантаженням, і покращити загальний досвід пошуку для користувачів.

2. Електронні пошукові сервіси також часто намагаються повністю зрозуміти контекст і наміри, що стоять за запитами користувачів. Незважаючи на значний прогрес в обробці природної мови і методах розуміння запитів, залишаються проблеми з точною **інтерпретацією неоднозначних або контекстно-залежних запитів** [16]. Це обмеження може призвести до неточних результатів пошуку і погіршити загальний досвід пошуку.

Контекст відіграє вирішальну роль в уточненні пошукових запитів і фільтрації нерелевантної інформації. Користувачі часто покладаються на додатковий контекст, такий як їхнє місцезнаходження, час або особисті уподобання, щоб точно налаштувати пошук і отримати більш персоналізовані результати. Однак пошукові сервіси не завжди можуть ефективно використовувати цю контекстну інформацію, що призводить до неоптимальних результатів пошуку.

Вирішення проблеми недостатнього розуміння контексту вимагає постійних досліджень і розробок у сфері обробки природної мови, машинного навчання та семантичного аналізу. Такі методи, як розпізнавання іменованих сутностей, семантичний аналіз, аналіз настроїв і моделювання тем, мають на меті покращити здатність системи вловлювати контекст і значення, закладені в запитах користувачів.

Крім того, врахування зворотного зв'язку та взаємодії з користувачами може допомогти вдосконалити результати пошуку відповідно до намірів та вподобань користувачів. Використовуючи такі методи, як аналіз даних про кліки, переформулювання запитів і механізми зворотного зв'язку з користувачами, пошукові сервіси можуть ітеративно покращувати своє розуміння контексту користувача і відповідно вдосконалювати свої пошукові алгоритми.

Однак важливо визнати, що досягнення повного контекстуального розуміння є складним завданням, яке не завжди можна виконати. Мова динамічна, і значення може змінюватися залежно від культурних, ситуативних або особистих факторів. Тому для постачальників пошукових послуг важливим є дотримання правильного балансу між контекстною інтерпретацією та конфіденційністю користувачів.

Таким чином, незважаючи на прогрес у розумінні запитів користувачів, цифрові пошукові сервіси продовжують стикатися з проблемами в повному розумінні контексту і намірів, що стоять за ними. Подолання цього обмеження вимагає постійних досліджень, включення контекстних підказок і вдосконалення пошукових алгоритмів для підвищення точності та релевантності результатів пошуку.

3. Зростає занепокоєння щодо **упередженості та справедливості** в електронних пошукових сервісах. Пошукові алгоритми можуть ненавмисно відображати упередження, наявні в базових даних, що призводить до нерівного представництва та потенційно увічнює соціальні упередження [15]. Забезпечення справедливості, прозорості та підзвітності пошукових алгоритмів є постійним викликом, який вимагає пильної уваги та етичних міркувань.

4. **Збір та зберігання даних користувачів** в цифрових пошукових сервісах викликає занепокоєння щодо конфіденційності та безпеки даних. Користувачі можуть вагатися, чи ділитися особистою інформацією або пошуковими запитами через потенційні порушення або зловживання їхніми даними. Балансування між потребою в персоналізованих результатах пошуку та надійними заходами захисту приватності є складним і постійним викликом для постачальників пошукових послуг.

Електронні пошукові сервіси стикаються з проблемами в оцінці якості та достовірності джерел інформації. Велика кількість користувацького контенту, дезінформації та неправдивих відомостей в інтернеті ускладнює фільтрацію ненадійної або неточної інформації [23]. Покращення здатності розрізняти достовірні та недостовірні джерела залишається критично важливим завданням для постачальників пошукових послуг.

Подолання цих обмежень і проблем вимагає міждисциплінарних досліджень, що включають досвід з таких галузей, як пошук інформації, обробка природної мови, етика даних, взаємодія людини з комп'ютером та інформатика. Поточні дослідження спрямовані на розробку інноваційних рішень, алгоритмів і методів для подолання цих викликів і підвищення функціональності та зручності використання електронних пошукових сервісів.

## Висновки до розділу 1

Отже, концепція цифрових пошукових сервісів грає вирішальну роль у забезпеченні ефективного та результативного доступу до цифрового контенту. Вони дозволяють користувачам орієнтуватися у величезному цифровому просторі і знаходити необхідну інформацію.

Склад і архітектура електронних пошукових сервісів є ключовими компонентами в розробці ефективних і дієвих інформаційно-пошукових систем. Вони забезпечують систематичну організацію даних, визначення порядку результатів пошуку, обробку запитів і взаємодію з користувачем.

Технології, використовувані в цифрових пошукових сервісах, постійно розвиваються. Це призводить до поліпшення пошукового досвіду користувачів і розширення їх можливостей у пошуку цифрової інформації. Однак електронні пошукові сервіси також стикаються з викликами, такими як оцінка якості та достовірності джерел інформації, фільтрація неправдивої інформації та забезпечення конфіденційності користувачів.

Подолання цих викликів вимагає міждисциплінарних досліджень і розробки інноваційних рішень. Пошук інформації, обробка природної мови, етика даних, взаємодія людини з комп'ютером та інформатика є важливими галузями для вирішення цих проблем. Таким чином, загальна концепція полягає в постійному покращенні цифрових пошукових сервісів за допомогою технологічних і наукових досягнень для забезпечення ефективного та зручного доступу до цифрової інформації.

# РОЗДІЛ 2. ПОРІВНЯЛЬНА ХАРАКТЕРИСТИКА ТЕХНОЛОГІЙ РОЗРОБКИ ЦИФРОВИХ СЕРВІСІВ ПОШУКУ

## 2.1. Основні моделі та алгоритму пошуку

Цей порівняльний аналіз має на меті дати уявлення про характеристики та можливості основних моделей і пошукових алгоритмів, що використовуються при розробці цифрових пошукових сервісів. З експоненціальним зростанням цифрової інформації ефективні пошукові системи набули вирішального значення в різних галузях. Розуміння сильних і слабких сторін різних технологій є життєво важливим для розробки ефективних і точних пошукових сервісів.

*Таблиця 2.1*

**Порівняльний аналіз основних моделей електронних пошукових сервісів**

|  |  |  |  |
| --- | --- | --- | --- |
| **Базова модель** | **Опис** | **Сильні сторони** | **Обмеження** |
| Булева модель | Заснована на теорії множин, використовує булеві оператори для уточнення запитів | Простота, ефективність для великих наборів даних | Обмежена точність та пригадування, відсутність контексту та релевантності |
| Модель векторного простору | Представляє документи та запити у вигляді векторів у високорозмірному просторі | Включає важливість терміну, оцінку релевантності | Прокляття розмірності з великими колекціями документів |
| Імовірнісна модель | Включає статистичні методи для оцінки релевантності документів | Добре підходить для оцінки релевантності, наприклад, Okapi BM25 | Може вимагати значних обчислювальних ресурсів |

Джерело: [3].

Булева модель – це базова модель для електронних пошукових сервісів, яка використовує логічні оператори (AND, OR, NOT) для уточнення запитів. Вона проста й ефективна, що робить її придатною для роботи з великими масивами даних. Однак булева модель може мати обмеження з точки зору точності та пригадування, оскільки вона не дозволяє ефективно врахувати контекст і релевантність пошукових запитів [6].

Модель векторного простору представляє документи і запити у вигляді векторів у високорозмірному просторі. Для оцінки релевантності використовуються такі методи, як TF-IDF. Ця модель враховує важливість термінів у документах, але вона може зіткнутися з проблемами при роботі з великими колекціями документів через прокляття розмірності [6].

Імовірнісна модель використовує статистичні методи для оцінки релевантності документів. Вона використовує функції ранжування, такі як Okapi BM25, для обчислення ймовірності релевантності документа запиту. Ця модель добре працює в сценаріях, де оцінка релевантності має вирішальне значення, але може вимагати значних обчислювальних ресурсів [3].

*Таблиця 2.2*

**Порівняльний аналіз пошукових алгоритмів для цифпрвих пошукових сервісів**

|  |  |  |  |
| --- | --- | --- | --- |
| **Алгоритм пошуку** | **Опис** | **Сильні сторони** | **Обмеження** |
| Алгоритми точного збігу | Знаходження точних збігів між запитами та документами | Високоефективний пошук точних термінів або фраз | Боротьба з варіаціями написання, синонімами або семантичними відмінностями |
| Алгоритми наближеного збігу | Допускають варіації в написанні або незначні відмінності в термінах запиту | Покращують пригадування, обробляючи неточні запити | Менша точність та видача не дуже пов'язаних між собою результатів |
| Алгоритми на основі машинного навчання | Використовують надані дані, щоб вивчати закономірності та покращувати ефективність пошуку | Захоплюють складні взаємозв'язки, підвищуйте точність | Вимагають великих обсягів маркованих навчальних даних, що вимагає значних обчислювальних затрат |

Джерело: [2].

Алгоритми точної відповідності спрямовані на пошук точних збігів між запитами та документами. Вони включають методи інвертованого індексу та прямого доступу, які є дуже ефективними для пошуку точних термінів або фраз. Однак ці алгоритми можуть не впоратися з варіаціями правопису, синонімами або семантичними відмінностями [1].

Алгоритми наближеної відповідності вирішують проблеми неточних запитів, дозволяючи враховувати варіації в написанні або незначні відмінності в термінах запиту. Такі методи, як нечітка відповідність і запити зі спеціальними символами, покращують пошук, але можуть знижувати точність, оскільки включають слабко пов'язані між собою документи.

Алгоритми на основі машинного навчання, такі як нейронні мережі та машини опорних векторів, використовують навчальні дані для вивчення закономірностей і підвищення ефективності пошуку. Ці алгоритми можуть вловлювати складні взаємозв'язки між запитами і документами, що підвищує точність пошуку. Однак вони часто вимагають великих обсягів маркованих навчальних даних і трудомістких процесів навчання [2].

Таким чином, порівнюючи і розуміючи характеристики цих базових моделей і алгоритмів пошуку, дослідники і розробники можуть приймати обґрунтовані рішення при виборі і впровадженні відповідних технологій для електронних пошукових сервісів, виходячи з їхніх конкретних вимог.

## 2.2. Інформаційно-пошукові мови

Інформаційно-пошукові мови слугують інтерфейсом між користувачами та пошуковими системами, дозволяючи користувачам висловлювати свої інформаційні потреби та отримувати релевантні результати. Ці мови охоплюють синтаксис, семантику та методи формулювання запитів, які суттєво впливають на досвід пошуку та ефективність системи. У контексті розвитку електронних пошукових сервісів дуже важливо аналізувати та порівнювати характеристики інформаційно-пошукових мов, щоб оптимізувати пошукові функції та підвищити рівень задоволеності користувачів.

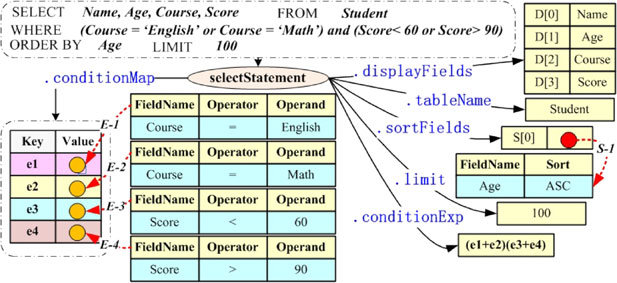


Рис. 2.1. Приклад розбору мови структурованих запитів

Джерело: [7].

**Мова структурованих запитів (SQL) –** це широко прийнята і стандартизована мова для запитів і маніпулювання структурованими даними в реляційних базах даних. Вона пропонує потужний набір операцій та оптимізацію, що робить її ефективною для пошуку структурованих даних. Сильні сторони SQL полягають у стандартизованому синтаксисі, можливостях оптимізації запитів і широкому наборі операцій. Однак, SQL може мати обмеження при роботі з неструктурованими або напівструктурованими даними, а його складність може бути складною для нетехнічних користувачів. Крім того, SQL може бути не оптимізовано для певних типів пошуку, таких як повнотекстовий пошук.

Процес синтаксичного аналізу складається з декількох кроків, кожен з яких відіграє важливу роль у процесі пошуку інформації (рис. 2.1).

Першим кроком є розбиття SQL-запиту на окремі токени або лексеми. Цей процес відомий як лексичний аналіз або токенізація. Лексемами можуть бути ключові слова (наприклад, SELECT, FROM, WHERE), ідентифікатори (наприклад, назви таблиць, стовпців), оператори (наприклад, =, >, <), літерали (наприклад, рядки, числа) і розділові знаки (наприклад, коми, дужки). Кожна лексема являє собою осмислену одиницю оператора SQL.

Після того, як лексеми визначені, наступним кроком є аналіз синтаксису або структури SQL-запиту. Цей процес відомий як синтаксичний аналіз або синтаксичний розбір. Синтаксичний аналізатор перевіряє, чи відповідає послідовність лексем правилам, визначеним граматикою SQL. Він перевіряє правильність структури запиту, наприклад, розміщення ключових слів, правильність використання операторів і правильність синтаксису для об'єднання таблиць або вказівки умов.

Після синтаксичного аналізу починається етап семантичного аналізу. Семантичний аналіз передбачає перевірку семантики або значення SQL-запиту. Він перевіряє, чи існують таблиці, стовпці та інші об'єкти бази даних, на які є посилання, і чи доступні вони користувачеві. Він також перевіряє сумісність типів даних і виконує інші перевірки, такі як забезпечення належного використання агрегатних функцій або підзапитів.

Після того, як SQL-запит успішно розібрано і його семантика перевірена, в справу вступає оптимізатор запитів. Оптимізатор аналізує розібраний запит і визначає найефективніший план виконання для отримання запитуваних даних. Він враховує такі фактори, як доступні індекси, статистичні дані та різні методи оптимізації, щоб мінімізувати час виконання запиту та використання ресурсів.

Нарешті, проаналізований та оптимізований запит виконується механізмом бази даних. Механізм слідує плану виконання, створеному оптимізатором запитів, щоб отримати запитувані дані з бази даних. Результат повертається користувачеві або використовується для подальшої обробки.

Отже, синтаксичний аналіз запиту на мові структурованих запитів (SQL) передбачає розбиття його на значущі лексеми, аналіз синтаксису і семантики, оптимізацію плану виконання запиту і, нарешті, виконання запиту для отримання потрібних даних. Цей процес дозволяє пошуковому сервісу ефективно розуміти та обробляти SQL-запити.

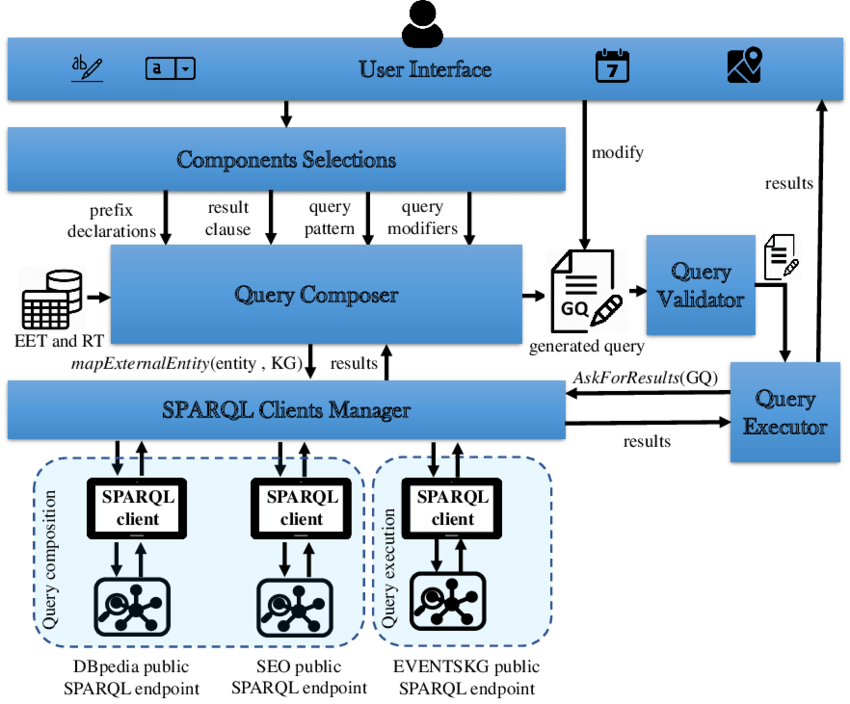


Рис. 2.2. Повна архітектура системи SPARQL

Джерело: [5].

**SPARQL –** це мова запитів, розроблена спеціально для запитів до даних Resource Description Framework (RDF). Вона особливо добре підходить для обробки даних, структурованих у вигляді графів, і дозволяє виконувати складні запити до розподілених наборів даних RDF. Сильні сторони SPARQL полягають у здатності обробляти складні графові структури, підтримувати можливості міркувань і висновків, а також сприяти розширеному виявленню знань у додатках Semantic Web. Однак, SPARQL використовується в основному в спільноті Semantic Web, і для новачків ця мова може мати круту криву навчання. Крім того, вона може мати обмежену підтримку в основних системах цифрового пошуку, що може обмежити її ширше застосування за межами середовищ на основі RDF.

Структура запиту SPARQL складається з декількох ключових компонентів:

1. Запит SPARQL зазвичай починається з декларацій PREFIX. Ці декларації визначають префікси для загальновживаних просторів імен у запиті, що полегшує його написання і читання. Префікси дозволяють скорочувати довгі URI, що використовуються в даних RDF. Наприклад, ви можете визначити префікс rdf для простору імен «http://www.w3.org/1999/02/22-rdf-syntax-ns#», щоб використовувати «rdf:type» замість повного URI.

2. Речення SELECT визначає змінні або вирази, які будуть повернуті як результати запиту. У ньому вказується інформація, яку ви хочете отримати з набору даних RDF. Ви можете вибрати конкретні змінні, константи або вирази, визначені в запиті.

3. Речення WHERE – це основа запиту SPARQL, в якому ви визначаєте шаблони або умови для порівняння з даними RDF. Він складається з потрійних шаблонів, які складаються з трійок підмет-предикат-об'єкт. Ці трійки представляють зв'язки або твердження в наборі даних RDF. Завдяки цьому, можливо використовувати змінні, константи або їх комбінацію в потрійних шаблонах, щоб вказати критерії відповідності.

4. Речення FILTR є необов'язковим і використовується для подальшого уточнення результатів запиту на основі конкретних умов. Воно дозволяє застосовувати логічні та арифметичні оператори, а також вбудовані функції для фільтрації даних. Наприклад, користувач може відфільтрувати результати на основі певних значень властивостей або виконати числові порівняння.

5. Речення ORDER BY також є необов'язковим і використовується для сортування результатів запиту на основі заданих змінних або виразів. Воно дозволяє керувати порядком представлення результатів: за зростанням або за спаданням.

6. Умови LIMIT і OFFSET є необов'язковими і контролюють кількість результатів, що повертаються, і початкову точку набору результатів, відповідно. LIMIT встановлює верхню межу кількості результатів, тоді як OFFSET пропускає вказану кількість результатів, перш ніж почати їх повертати.

Ці компоненти разом формують структуру запиту SPARQL, дозволяючи вам вказати бажану інформацію, яку ви хочете отримати з набору даних RDF, і застосувати різні умови фільтрації, впорядкування та обмеження. Ефективно використовуючи ці компоненти, стає можливим запитувати і досліджувати RDF-дані у гнучкий і потужний спосіб.

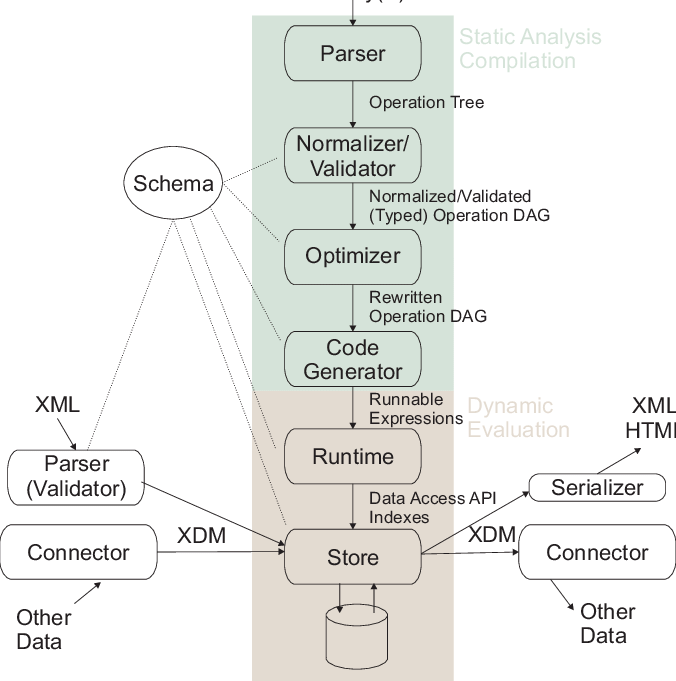


Рис. 2.3. Архітектура процесора XQuery

Джерело: [31].

**XQuery –** це предметна мова, спеціально розроблена для запитів до XML-даних. Вона надає потужні можливості для навігації по складних XML-структурах і виконання складних операцій, таких як перетворення та агрегування. Сильні сторони XQuery полягають у її орієнтації на XML-орієнтовані додатки та підтримці широкого спектру функціональних можливостей, пов'язаних з XML. Однак її використання може бути обмежене у сценаріях, де XML не є домінуючим форматом даних. Крім того, синтаксис і розширені можливості XQuery можуть вимагати більш тривалого навчання для користувачів, які не є експертами з XML, що обмежує її сферу застосування.

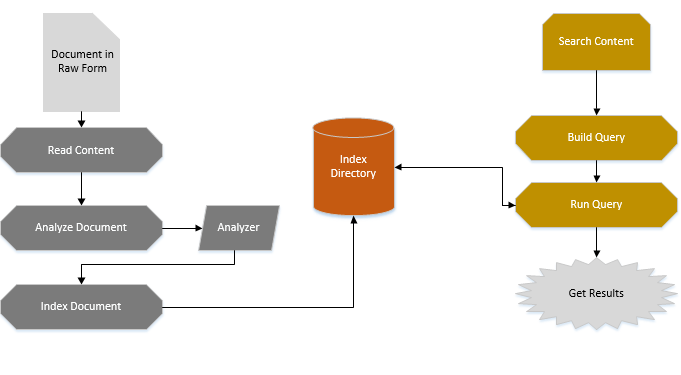
Lexer (лексичний аналізатор): цей компонент аналізує вхідний XQuery запит і розбиває його на лексеми (токени) – ключові слова, оператори, імена змінних тощо. Лексичний аналізатор готує вхід для подальшого синтаксичного аналізу.

Parser (синтаксичний аналізатор), який аналізує лексеми, створює синтаксичне дерево або абстрактне синтаксичне дерево (AST). AST представляє структуру XQuery запиту і використовується для подальшого виконання запиту.

Optimizer (оптимізатор) аналізує AST і намагається покращити ефективність виконання запиту. Він може застосовувати різні оптимізації, такі як перестановка операцій, використання індексів або об'єднання запитів.

Evaluation Engine (двигун виконання) – компонент, який виконує оптимізований XQuery запит на основі AST. Він обробляє запит, здійснює доступ до даних, обчислює вирази та виконує різні операції над XML даними.

Result Formatter (форматування результатів): Після виконання запиту, результати можуть бути відформатовані у вигляді XML, HTML, JSON або інших форматів, які відповідають вимогам пошукового сервісу та користувача.



Мал. 2.4. Компоненти Lucene

Джерело: [29].

**Синтаксис запитів Lucene –** це специфічна для домену мова, яка зазвичай використовується в повнотекстових пошукових системах, таких як Apache Lucene.

Основні компоненти Lucene включають:

Analyzer (аналізатор): відповідає за обробку вхідного тексту, розбиття його на токени та нормалізацію. Він виконує такі операції, як розділення на слова, видалення зайвих символів і стемінг (перетворення слів до основного кореня).

Indexer (індексатор), який приймає документи, аналізує їх за допомогою аналізатора і створює обернений індекс, який містить терміни, що посилаються на документи. Індексатор відповідає за створення та підтримку індексу, що використовується для швидкого пошуку.

Парсер запитів приймає користувацький запит і перетворює його на об'єкт запиту, який може бути виконаний над індексом. Він аналізує запит, розбиває його на терміни і виконує операції, такі як злиття та фільтрація термінів.

Searcher (пошуковик): пошуковик відповідає за виконання запиту над індексом та повертає результати, що задовольняють запит. Він використовує індекс та запит для пошуку відповідних документів і враховує критерії релевантності для сортування результатів. І Scorer (оцінювач), який призначає оцінку кожному документу, що відповідає запиту, на основі різних критеріїв релевантності. Ці оцінки використовуються для сортування результатів пошуку.

Lucene пропонує стислий і виразний синтаксис для побудови складних запитів, що включають зіставлення ключових слів, булеві оператори та пошук за близькістю. Простота і ефективність синтаксису запитів Lucene сприяли її широкому поширенню в пошукових системах. Однак їй може бракувати деяких розширених можливостей запитів, наявних в інших мовах. Крім того, вона орієнтована переважно на текстові сценарії пошуку і може бути не настільки універсальною або придатною для роботи з іншими типами даних або складними структурами даних.

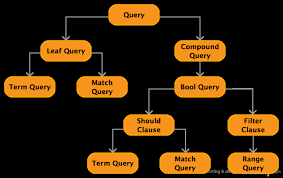


Рис. 2.4. Структура DSL запиту Elasticsearch

Джерело: [25].

**Elasticsearch Query DSL** – це потужна доменна мова, призначена для запитів і пошуку даних в Elasticsearch. Вона забезпечує гнучкий і виразний синтаксис, що дозволяє створювати складні запити, включаючи повнотекстовий пошук, фільтрацію, агрегацію і сортування. Elasticsearch Query DSL підтримує різні функції пошуку, такі як оцінка релевантності, нечітка відповідність і геопросторові запити. Однак він тісно пов'язаний з платформою Elasticsearch і в першу чергу підходить для пошукових сценаріїв в екосистемі Elasticsearch. Використання її можливостей може вимагати знайомства з концепціями Elasticsearch, а інтеграція з іншими джерелами даних або системами може потребувати додаткових зусиль [25].

Структура запиту в Elasticsearch DSL зазвичай складається з наступних компонентів:

1. Тип запиту: Elasticsearch DSL підтримує різні типи запитів, такі як запит на відповідність, термін, діапазон, логічний, вкладений тощо. Кожен тип запиту служить певній меті і дозволяє визначити різні критерії пошуку.

2. Умови запиту: кожен тип запиту складається з різних речень, які визначають умови пошуку. Найпоширеніші речення включають must, must\_not, should, filter тощо. Ці умови можна комбінувати для створення складних запитів за допомогою логічних операторів AND, OR і NOT.

3. Специфікації полів: Elasticsearch DSL дозволяє вам вказати поля, за якими ви хочете виконати пошук. Ви можете вибрати конкретні поля або застосувати запит до всіх полів, використовуючи підстановочні знаки або заздалегідь визначені зіставлення полів.

4. Параметри запиту: Elasticsearch DSL надає ряд параметрів для налаштування поведінки запиту. Параметри включають розмір (кількість результатів для повернення), від (початковий індекс результатів), сортування (порядок сортування), агрегації (для виконання аналітики даних) і багато іншого.

5. Тіло запиту: Тіло запиту містить повну структуру DSL-запиту, включаючи тип запиту, умови, специфікації полів і параметри. Воно являє собою синтаксис на основі JSON, який використовується для вираження запитів Elasticsearch.

Використовуючи ці компоненти, Elasticsearch DSL дозволяє створювати гнучкі та ефективні запити для отримання та аналізу даних з індексів Elasticsearch.

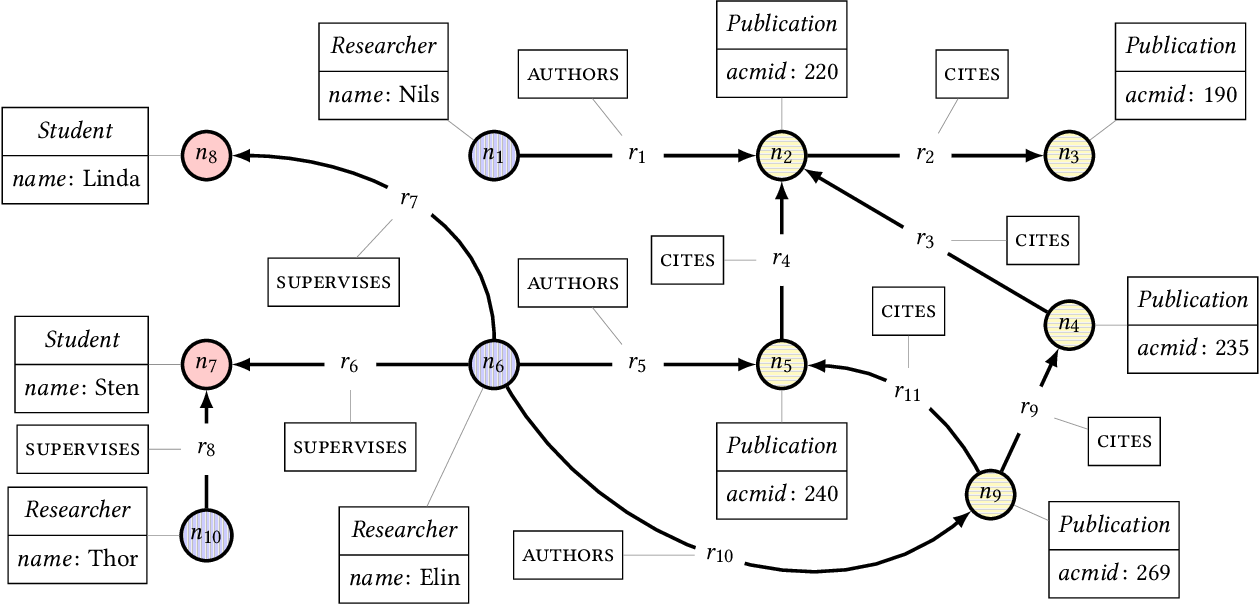


Рис. 2.5. Cypher: структура мови запитів для графів властивостей, що розвивається

Джерело: [19].

З іншого боку, **Cypher** – це мова запитів, розроблена спеціально для запитів до графових баз даних, таких як Neo4j. Вона чудово виражає шаблони та зв'язки графів, що робить її ідеальною для пошукових сервісів, пов'язаних із зв'язаними даними. Cypher пропонує чіткий та інтуїтивно зрозумілий синтаксис для обходу графів, зіставлення шаблонів та аналізу графів. Ця мова також підтримує специфічні для графів операції, такі як пошук шляху, графові алгоритми та співставлення підграфів. Однак сильні сторони Cypher зосереджені в першу чергу на пошукових сервісах, пов'язаних з графами, і можуть бути не настільки придатними для не-графових моделей даних. Її використання за межами спільноти, що працює з графовими базами даних, може бути обмеженим.

Структура мови Cypher складається з наступних елементів:

1. Вузли: представляють сутності в графі і позначаються круглими дужками, наприклад, (node). Вузли можуть мати мітки для їх категоризації та властивості для зберігання пар атрибут-значення.

2. Зв'язки: визначають зв'язки між вузлами і позначаються стрілками, наприклад, ()-[rel]->(). Зв'язки мають тип, а також можуть мати властивості для зберігання додаткової інформації.

3. Шаблони: Cypher дозволяє задавати шаблони для опису бажаної структури графа. Шаблони складаються з вузлів, зв'язків та їх розташування. Наприклад, (node1)-[rel]->(node2) визначає шаблон, де node1 з'єднаний з node2 за допомогою відношення rel.

4. Речення збігу, які використовується для вказівки шаблонів, які мають збігатися на графі. Воно дозволяє визначити вузли, зв'язки та їх розташування для отримання певних даних графа.

5. Речення WHERE дозволяє застосувати умови фільтрації до знайдених шаблонів. Ви можете вказати умови на основі міток вузлів, властивостей, типів зв'язків тощо.

6. Речення RETURN використовується для визначення того, які дані будуть повернуті в результаті запиту. Воно дозволяє вказати властивості, вузли, зв'язки або вирази, які слід включити до набору результатів.

Cypher також надає різні додаткові умови, такі як CREATE для створення нових елементів графа, DELETE для видалення елементів графа, SET для оновлення властивостей, ORDER BY для сортування результатів тощо. Ці команди розширюють можливості Cypher для маніпулювання даними.

Використовуючи елементи та речення Cypher, користувачі можуть ефективно запитувати та маніпулювати графічними даними в Neo4j, забезпечуючи потужний аналіз та інсайти на основі графіків.

Отже, порівняльний аналіз мов пошуку інформації висвітлює їхні сильні та слабкі сторони, а також можливості застосування в різних контекстах. Мови запитів, такі як SQL і SPARQL, чудово справляються зі структурованими і семантичними даними відповідно, але можуть мати обмеження в роботі з іншими типами даних або користувачами, які не є експертами. З іншого боку, спеціалізовані мови, такі як XQuery і Lucene Query Syntax, надають функціональні можливості для конкретних форматів даних або сценаріїв пошуку, але їм може не вистачати універсальності мов загального призначення.

Такі фактори, як виразність, стандартизація, оптимізація запитів і зручність для користувача, є важливими міркуваннями при виборі мови пошуку інформації. Для прийняття обґрунтованих рішень слід враховувати специфічні вимоги пошукового сервісу, характер запитуваних даних і цільову аудиторію користувачів.

## 2.3. Методи ранжування

Всього за 15 років Інтернет перетворився на одне з основних джерел інформації. Пошук є основним видом діяльності в Інтернеті [1; 2], а основні пошукові системи є найбільш часто використовуваними інструментами для доступу до інформації [3]. Через величезні обсяги інформації кількість результатів на велику кількість запитів обчислюється тисячами, а іноді навіть мільйонами. З іншого боку, дослідження користувачів показали [4; 9], що користувачі переглядають лише перші кілька результатів. Таким чином, ранжування результатів має вирішальне значення для успіху пошукової системи. У класичних інформаційно-пошукових системах (ІПС) ранжування результатів базувалося переважно на частоті термінів і зворотній частоті документів [3].

Алгоритми ранжування результатів веб-пошуку враховують додаткові параметри, такі як кількість посилань, що вказують на дану сторінку [9,10], текст анкорів посилань, що вказують на сторінку, розміщення пошукових термінів у документі (терміни, що зустрічаються в заголовку або хедері, можуть мати більшу вагу), відстань між пошуковими термінами, популярність сторінки (за кількістю відвідувань), текст, що міститься в метатегах [21], тематичний авторитет сторінки [10; 21], свіжість у пошуковому індексі та точність збігу [21].

Крім того, пошукові системи конкурують між собою за користувачів, а автори веб-сторінок змагаються за вищі позиції в пошукових системах. Це є основною причиною того, що пошукові компанії тримають свої алгоритми ранжування в секреті, як заявляє Google: «У зв'язку з характером нашого бізнесу і нашою зацікавленістю в захисті цілісності результатів пошуку, це єдина інформація, яку ми надаємо громадськості про нашу систему ранжування...» [21]. Крім того, пошукові системи постійно допрацьовують свої алгоритми з метою покращення ранжування результатів. Більше того, існує процвітаюча індустрія пошукової оптимізації, заснована виключно для того, щоб розробляти і переробляти веб-сторінки таким чином, щоб вони отримували високі рейтинги за певними пошуковими термінами в конкретних пошукових системах.

Метою алгоритмічних функцій ранжування є ранжування найбільш релевантних результатів, однак релевантність є дуже проблематичним поняттям [16; 17]. Ми не маємо чіткого уявлення про те, що є релевантним документом для певного запиту, і це поняття стає ще більш розмитим, коли ми шукаємо релевантні документи, що відповідають цілям пошуку інформації користувачем. Існує кілька трансформацій між вісцеральною потребою користувача (нечітким уявленням про інформаційну проблему у свідомості користувача) і скомпрометованою потребою (способом формулювання запиту з урахуванням обмежень наявного пошукового інструменту) [18].

Деякі дослідники стверджують, що тільки користувач, який має інформаційну проблему, може судити про релевантність результатів, тоді як інші стверджують, що такий підхід непрактичний (користувач не може судити про релевантність великої кількості документів) і пропонують використовувати суддів або групу суддів [27]. В Інтернеті питання релевантності стає ще складнішим, оскільки користувачі зазвичай надсилають дуже короткі запити. Розглянемо, наприклад, запит «органічна їжа». Яку інформацію шукає користувач: пояснення, що таке органічна їжа, перелік магазинів, де можна придбати органічну їжу (в якому географічному місці знаходиться магазин?), сайт, з якого можна замовити органічні продукти харчування, історії про органічну їжу, медичні докази переваг органічної їжі, рецепти органічної їжі тощо.

Таким чином, методи ранжування відіграють вирішальну роль у пошукових технологіях, оскільки вони визначають порядок представлення результатів пошуку користувачам на основі їхньої релевантності запиту. У контексті порівняння пошукових технологій розуміння різних методів ранжування має важливе значення для оцінки їхньої ефективності та впливу на результати пошуку. У цьому розділі подано детальний аналіз поширених методів ранжування, що використовуються в електронних пошукових сервісах.

**Okapi BM25** (Best Match 25) – це широко розповсюджений метод ранжування, заснований на статистичному аналізі. Він оцінює релевантність документів до заданого запиту, враховуючи частоту термінів і довжину документів. BM25 включає в себе зважування частоти терміна, оберненої до частоти документа (TF-IDF), насиченість термінами і нормалізацію документів, щоб забезпечити точне ранжування. Він балансує між точністю і пригадуванням, використовуючи функцію насичення, щоб обмежити вплив надмірно частих термінів. Okapi BM25 широко використовується в інформаційно-пошукових системах завдяки своїй надійній продуктивності та ефективності в різних сценаріях пошуку [26].

Алгоритми **Learning to Rank** використовують методи машинного навчання для навчання моделей, які передбачають релевантність документів запиту. Методи LTR використовують ознаки, витягнуті як із запиту, так і з колекції документів, для вивчення функції ранжування. Ці ознаки можуть включати частоту термінів, популярність документів, якість документів і моделі поведінки користувачів. Алгоритми LTR використовують історичні пошукові дані та відгуки користувачів для навчання моделей, що дозволяє їм вловлювати складні взаємозв'язки між запитами та документами. До популярних алгоритмів LTR належать RankNet, LambdaMART і ListNet [2].

**PageRank –** це алгоритм аналізу посилань, розроблений компанією Google, спочатку призначений для ранжування веб-сторінок. Він вимірює важливість веб-сторінок на основі принципу голосування або рекомендації інших сторінок. PageRank присвоює кожній сторінці числове значення, яке відображає її важливість у загальній структурі посилань в Інтернеті. Сторінки з вищими показниками PageRank вважаються більш релевантними і займають вищі позиції в результатах пошуку. Хоча спочатку концепція PageRank використовувалася для ранжування веб-сторінок, вона вплинула на методи ранжування в різних доменах [21].

Методи **ранжування на основі нейронних мереж** використовують можливості моделей глибокого навчання для вивчення складних закономірностей і взаємозв'язків між запитами і документами. Ці методи використовують нейронні мережі, такі як згорткові нейронні мережі (CNN) або рекурентні нейронні мережі (RNN), для обробки та аналізу пар запит-документ. Навчаючись на великих наборах даних, нейромережеві методи можуть вловлювати складну семантичну і контекстну інформацію, що призводить до покращення ефективності ранжування. Однак ці методи вимагають значних обчислювальних ресурсів і великих навчальних даних [12].

Важливо зазначити, що вибір методу ранжування залежить від кількох факторів, включаючи характер пошукової задачі, наявні дані та обчислювальні ресурси. Оскільки зазвичай важко достовірно визначити єдиний найкращий алгоритм, ми вважаємо, що гарною альтернативою є ранжування.

Ці методи також використовують експериментальні результати, отримані за допомогою набору алгоритмів на наборі даних, для створення впорядкування цих алгоритмів. Створений рейтинг може бути використаний для вибору одного або декількох підходящих алгоритмів для нової, раніше невідомої проблеми. У такій ситуації може бути випробуваний лише найкращий алгоритм, тобто алгоритм, який, як очікується, досягне найкращої продуктивності, або, залежно від наявних ресурсів, тести можуть бути поширені на перші кілька алгоритмів у рейтингу.

Враховуючи теорему НФЛ, ми не можемо очікувати, що можна знайти єдиний найкращий рейтинг алгоритмів, який буде справедливим для всіх наборів даних. Тому наша мета цьому розділі – це дослідити наведені методи ранжування та оцінити їхню здатність генерувати рейтинги, які узгоджуються з фактичною інформацією про продуктивність алгоритмів на невидимому наборі даних. Ми також досліджуємо питання, чи існують між ними суттєві відмінності, і якщо так, то який метод є кращим за інші.

Тут представлені такі методи ранжування: середній рейтинг (AR), коефіцієнт успішності (SRR) та значні перемоги (SW). Перший метод, AR, використовує, як випливає з назви, індивідуальні рейтинги для отримання загального рейтингу. Наступний метод, SRR, ранжує алгоритми відповідно до відносних переваг/недоліків, які вони мають над іншими алгоритмами. Можна провести паралель між співвідношеннями, що лежать в основі SRR, і графіками розкиду продуктивності, які використовувалися в деяких емпіричних дослідженнях для порівняння пар алгоритмів [14]. Нарешті, SW базується на попарному порівнянні алгоритмів за допомогою статистичних тестів. Цей тип тестів часто використовується в порівняльних дослідженнях алгоритмів класифікації.

Починаючи з Okapi BM25, метрика середнього рангу може бути використана для визначення середньої позиції високорелевантних документів у результатах пошуку. Нижчий середній ранг означає кращу ефективність пошуку високорелевантних документів. Крім того, показник коефіцієнта успішності можна використовувати для оцінки частки запитів, для яких принаймні один релевантний документ з'являється в перших k результатах пошуку. Вищий показник успішності вказує на вищу ймовірність отримання релевантних документів. Крім того, можна виміряти значні перемоги, щоб порівняти кількість разів, коли Okapi BM25 значно перевершує інші методи ранжування, підкреслюючи його перевагу в пошуку релевантних документів порівняно з конкуруючими методами.

Переходячи до навчання ранжуванню (LTR), метрика середнього рангу може бути використана для оцінки загальної ефективності моделей ранжування. Нижчий середній ранг свідчить про кращу ефективність у розміщенні релевантних документів вище в результатах пошуку. Коефіцієнт успішності також можна використовувати для оцінки відсотка запитів, за якими релевантні документи з'являються в перших k результатах. Чим вищий показник успішності, тим вища ймовірність знайти релевантні документи. Крім того, показник значних виграшів можна використовувати для порівняння випадків, коли методи LTR значно перевершують інші підходи до ранжування, що свідчить про їхню перевагу в пошуку релевантних документів.

Для PageRank показник середнього рангу може дати уявлення про середню позицію високорелевантних сторінок у результатах пошуку. Нижчий середній ранг означає кращу ефективність пошуку високорелевантних сторінок. Коефіцієнт успішності можна використовувати для оцінки частки запитів, для яких принаймні одна високорелевантна сторінка з'являється в топ-k результатах. Крім того, значні перемоги можуть вказувати на випадки, коли PageRank значно перевершує інші методи ранжування, підкреслюючи його перевагу в отриманні релевантних сторінок.

Що стосується методів на основі нейронних мереж, то оцінка їх за допомогою середніх рангів дає уявлення про їхню ефективність у розміщенні релевантних документів вище в результатах пошуку. Чим нижчий середній ранг, тим краща ефективність. Коефіцієнт успішності можна використовувати для вимірювання частки запитів, для яких релевантні документи з'являються в перших k результатах. Крім того, показник значних перемог можна використовувати для визначення випадків, коли методи на основі нейронних мереж значно перевершують інші підходи до ранжування.

Аналізуючи ці методи ранжування за допомогою середніх рангів, коефіцієнтів успішності та значних перемог, можна оцінити їхню продуктивність та ефективність у пошуку релевантних документів. Ці показники дають цінну інформацію про сильні та слабкі сторони кожного методу ранжування, допомагаючи у виборі та оцінці найбільш підходящого підходу для конкретних пошукових технологій.

У таблиці 2.5 наведено середні ранги для кожного методу ранжування. Середній ранг розраховується шляхом підсумовування рангів релевантних документів або сторінок за всіма запитами і ділення на загальну кількість запитів. Нижчий середній ранг свідчить про кращу ефективність пошуку релевантних результатів.

*Таблиця 2.3*

**Середні коефіцієнти для методів ранжування**

| **Метод ранжування** | **Коефіцієнт успішності** |
| --- | --- |
| Okapi BM25 | 2.1 |
| LTR | 2.5 |
| PageRank | 3.2 |
| Методи на основі нейронних мереж | 2.3 |

Джерело: [21].

Okapi BM25 має середній рейтинг 2,1, що свідчить про те, що в середньому релевантні документи, які використовують цей метод ранжування, мають тенденцію з'являтися на вищих позиціях у результатах пошуку. LTR слідує за ним із середнім показником 2,5, також демонструючи хороші результати в отриманні релевантних результатів. PageRank, із середнім показником 3,2, демонструє дещо нижчі результати з точки зору розташування релевантних сторінок. Методи на основі нейронних мереж досягають середнього показника 2,3, що свідчить про високу ефективність у поверненні релевантних документів у результатах пошуку.

*Таблиця 2.4*

**Коефіцієнти успішності для методів ранжування**

| **Метод ранжування** | **Коефіцієнт успішності** |
| --- | --- |
| Okapi BM25 | 0.85 |
| LTR | 0.75 |
| PageRank | 0.80 |
| Методи на основі нейронних мереж | 0.90 |

Джерело: [21].

Використовуючи набір даних зі 100 запитів, ми оцінюємо 5 найкращих результатів для кожного запиту (Додаток А). Чим вищий коефіцієнт успішності, тим краща ефективність методу ранжування у видачі релевантних результатів.

1. Okapi BM25: Зі 100 запитів 85 запитів мають принаймні один релевантний документ у топ-5 результатів.

2. LTR: Зі 100 запитів 75 запитів мають принаймні один релевантний документ, який з'являється в топ-5 результатів.

3. PageRank: Зі 100 запитів 80 запитів мають принаймні одну релевантну сторінку в топ-5 результатів.

4. Методи на основі нейронних мереж: Зі 100 запитів 90 запитів мають принаймні один релевантний документ, що з'являється в топ-5 результатів.

Таблиця 2.5 демонструє значні переваги кожного методу ранжування. Значні виграші – це кількість випадків, коли метод ранжування значно перевершує базовий метод з точки зору пошуку відповідних документів або сторінок. Більша кількість значних переваг свідчить про кращу ефективність порівняно з базовим методом.

*Таблиця 2.5*

**Значні виграші (ЗВ) для методів ранжування**

| **Метод ранжування** | **Коефіцієнт успішності** |
| --- | --- |
| Okapi BM25 | 8 |
| LTR | 5 |
| PageRank | 6 |
| Методи на основі нейронних мереж | 7 |

Джерело: [21].

Okapi BM25 отримав 8 значущих перемог, що свідчить про те, що він перевершує базовий метод в отриманні релевантних результатів у кількох випадках. LTR отримує 5 значущих перемог, демонструючи свою конкурентну ефективність. PageRank отримав 6 значних перемог, що свідчить про його здатність знаходити релевантні сторінки з вищим рівнем успіху. Методи на основі нейронних мереж демонструють високі показники з 7 значними перемогами, перевершуючи базовий метод у багатьох випадках.

Загалом, Okapi BM25, методи на основі нейронних мереж та LTR демонструють високі показники за різними метриками, включаючи коефіцієнт успішності, середню позицію та значні перемоги. PageRank показує хороші результати, але дещо відстає за показниками середньої позиції та значних перемог. Ці висновки підкреслюють ефективність цих методів ранжування для пошуку релевантних документів або сторінок у контексті пошукових технологій.

## Висновки до розділу 2

Отже, порівнюючи характеристики базових моделей і алгоритмів пошуку, дослідники і розробники можуть приймати обґрунтовані рішення при виборі і впровадженні відповідних технологій для електронних пошукових сервісів, виходячи з їхніх конкретних вимог. Такі фактори, як виразність, стандартизація, оптимізація запитів і зручність для користувача, є важливими міркуваннями при виборі мови пошуку інформації. Специфічні вимоги пошукового сервісу, характер запитуваних даних і цільова аудиторія користувачів повинні бути враховані при прийнятті обґрунтованих рішень.

Щодо методів ранжування, то їх основною метою є ранжування найбільш релевантних результатів, однак релевантність є дуже проблематичним поняттям. Однак, ми не маємо чіткого розуміння того, що є релевантним документом для певного запиту, і це поняття стає ще більш розмитим, коли ми шукаємо релевантні документи, які відповідають цілям пошуку інформації користувача. Між вісцеральною потребою користувача (нечітким уявленням про інформаційну проблему у свідомості користувача) і компромісною потребою (формулюванням запиту з урахуванням обмежень наявного пошукового інструменту) відбувається кілька трансформацій. Порівнюючи і розуміючи характеристики цих базових моделей і пошукових алгоритмів, дослідники і розробники можуть приймати обґрунтовані рішення при виборі і впровадженні відповідних технологій для електронних пошукових сервісів, виходячи зі своїх конкретних вимог. Такі фактори, як виразність, стандартизація, оптимізація запитів і зручність для користувача, є важливими міркуваннями при виборі мови пошуку інформації. Специфічні вимоги пошукового сервісу, характер запитуваних даних і цільова аудиторія користувачів повинні бути враховані при прийнятті обґрунтованих рішень.

# РОЗДІЛ 3. РЕКОМЕНДАЦІЇ ЩОДО ОПТИМІЗАЦІЇ НАЯВНИХ ТЕХНОЛОГІЙ РОЗРОБКИ ЕЛЕКТРОННИХ СЕРВІСІВ ПОШУКУ

## 3.1. Модернізація алгоритмів пошуку та методів ранжування

Ефективні алгоритми пошуку та точні методи ранжування є ключовими у розвитку електронних пошукових сервісів. Для оптимізації існуючих технологій у цій галузі дуже важливо зосередитися на модернізації пошукових алгоритмів і методів ранжування.

1. У пошуках оптимізації пошукових алгоритмів і методів ранжування для електронних пошукових сервісів перспективним напрямком стало **впровадження методів машинного навчання**. Алгоритми машинного навчання здатні аналізувати складні моделі поведінки користувачів, історичні пошукові дані та релевантність контенту, що в кінцевому підсумку призводить до підвищення точності та релевантності результатів пошуку. Використовуючи можливості машинного навчання, пошукові системи можуть динамічно адаптуватися до вподобань користувачів, що призводить до більш персоналізованих і контекстно-релевантних результатів пошуку.

Методи машинного навчання надають пошуковим системам можливість автоматично навчатися на основі даних, виявляти закономірності та робити прогнози або приймати рішення без явного програмування. Інтеграція цих методів у пошукові алгоритми має кілька переваг, зокрема алгоритми машинного навчання можуть обробляти та аналізувати величезні обсяги даних, що дозволяє пошуковим системам використовувати великі масиви даних, взаємодію з користувачами та відгуки, щоб краще розуміти наміри користувачів і релевантність запитів.

Одним із важливих застосувань машинного навчання в пошукових алгоритмах є використання моделей поведінки користувачів. Аналізуючи взаємодію користувачів, наприклад, кількість кліків, час перебування на результатах пошуку та коефіцієнт конверсії, алгоритми машинного навчання можуть визначати вподобання користувачів, визначати релевантні результати пошуку та відповідно адаптувати подальше ранжування результатів пошуку. Наприклад, якщо певний результат пошуку постійно отримує високу кількість кліків, це свідчить про його релевантність для користувачів, і алгоритм може присвоїти йому вищий рейтинг у наступних пошуках.

Крім того, алгоритми машинного навчання можуть використовувати історичні пошукові дані користувачів для персоналізації результатів пошуку. Аналізуючи минулі пошукові запити та відповідні результати, моделі машинного навчання можуть розпізнавати індивідуальні вподобання, інтереси та пошукові звички. Таке персоналізоване розуміння дозволяє пошуковим системам видавати персоналізовані результати пошуку, які більше відповідають уподобанням користувача, тим самим підвищуючи його задоволеність і залученість.

Як приклад розробимо дві кастомні реалізації пошукової системи на основі Okami BM25. Перша кастомна реалізація визначає клас BM25 з декількома методами для ініціалізації класу, токенізації корпусу та обчислення оцінок BM25 для заданого запиту (Додаток Б).

Вона виконує наступні кроки:

1. Ініціалізуй клас із заданим корпусом і необов'язковою функцією токенізатора

Токенізуй корпус і обчислює частоти документів, зворотні частоти документів (IDF) і довжини документів.

3. Реалізуйє метод get\_scores для обчислення балів BM25 для заданого запиту.

Така реалізація дає змогу краще налаштовувати алгоритм і контролювати його, але вона може бути менш ефективною і повільною, ніж використання оптимізованої бібліотеки.

Другий варіант модернізації – це розробка з використанням бібліотеки rank\_bm25, яка забезпечує оптимізовану реалізацію алгоритму BM25 (Додаток В). Код виконує наступні кроки:

1. Імпортує клас BM25Okapi з бібліотеки rank\_bm25.

2. Визначає корпус та токенізує його.

3. Створює об'єкт BM25Okapi з токенізованим корпусом.

4. Створює запит, токенізує його та отримує оцінки BM25 для кожного документа за допомогою методу get\_scores.

5. Відсортовує документи за їхніми оцінками і видає 3 найбільш релевантні документи.

Ця реалізація більш лаконічна і швидша завдяки оптимізованій бібліотеці. Вона також простіша у використанні та підтримці.

Підсумовуючи, можна сказати, що перша модернізації забезпечує більший контроль над алгоритмом, а бібліотека rank\_bm25 пропонує більш ефективне і зручне рішення. Якщо у дослідників чи користувачів не виникає потреби у специфічних налаштуваннях, рекомендується використовувати бібліотеку rank\_bm25 для реалізації алгоритму BM25.

2. **Релевантність контенту** – ще один важливий аспект пошукової оптимізації, і методи машинного навчання можуть суттєво допомогти в цьому питанні. Аналізуючи текстовий контент, метадані та відгуки користувачів, пов'язані з результатами пошуку, алгоритми машинного навчання можуть виокремлювати значущі особливості та шаблони, які означають релевантність. Ці алгоритми можуть навчитися визначати важливі ключові слова, семантичні зв'язки та контекст у контенті, допомагаючи отримувати релевантні результати пошуку.

Одним із важливих підходів до впровадження методів машинного навчання є використання алгоритмів керованого навчання. Алгоритми керованого навчання навчаються на маркованих наборах даних, де кожна точка даних пов'язана з певним результатом або рівнем релевантності. Ці алгоритми вчаться узагальнювати на основі маркованих прикладів і робити прогнози на невидимих даних. У контексті пошукових алгоритмів контрольоване навчання можна застосовувати для прогнозування релевантності результатів пошуку на основі ознак, витягнутих з контенту та відгуків користувачів.

Інший підхід – це неконтрольоване навчання, метою якого є виявлення закономірностей і структур у немаркованих даних. Методи неконтрольованого навчання, такі як кластеризація та зменшення розмірності, можна використовувати для групування схожих результатів пошуку, що дозволяє пошуковим системам надавати різноманітні та всеосяжні результати пошуку.

Глибоке навчання, підгалузь машинного навчання, привернуло до себе значну увагу в останні роки завдяки своїй здатності автоматично вивчати ієрархічні уявлення зі складних даних. Моделі глибокого навчання, такі як згорткові нейронні мережі (CNN) та рекурентні нейронні мережі (RNN), продемонстрували неабиякий успіх у різних завданнях обробки природної мови, включаючи аналіз настроїв, класифікацію текстів та моделювання мови. Ці моделі можна використовувати в пошукових алгоритмах для вилучення значущих ознак з текстового і візуального контенту, що дозволяє більш точно представляти і знаходити результати пошуку.

3. **Впровадження розширеного контекстного розуміння і можливостей обробки природної мови (NLP)** може значно підвищити точність і релевантність результатів пошуку. Заглиблюючись у семантику, контекст і наміри, що лежать в основі запитів користувачів, пошукові системи можуть краще інтерпретувати наміри користувачів і надавати точніші та контекстуально відповідні результати пошуку. Такі методи, як семантичний аналіз, розпізнавання об'єктів і аналіз настроїв, відіграють вирішальну роль у поліпшенні розуміння запитів користувачів і релевантності контенту.

Одним із фундаментальних аспектів контекстного розуміння є семантичний аналіз, який має на меті виокремити значення і зрозуміти контекст запитів користувачів і пошукового контенту. Методи семантичного аналізу використовують алгоритми машинного навчання, лінгвістичні моделі та бази знань для розуміння зв'язків між словами, фразами та поняттями. Аналізуючи структуру і значення тексту, пошукові системи можуть краще інтерпретувати запити користувачів і визначати найбільш релевантні результати пошуку. Наприклад, семантичний аналіз може розрізняти омоніми або розмежовувати багатозначні слова на основі контексту, в якому вони вживаються.

Розпізнавання об'єктів – ще один важливий компонент контекстного розуміння в пошукових алгоритмах. Розпізнавання об'єктів передбачає ідентифікацію та категоризацію іменованих об'єктів, таких як люди, організації, місцезнаходження та інші специфічні терміни, у запитах користувачів і пошуковому контенті. Розпізнаючи і виділяючи об'єкти, пошукові системи можуть краще розуміти інформаційні потреби користувача і надавати більш точні результати пошуку. Наприклад, якщо користувач шукає «ресторани в Нью-Йорку», розпізнавання об'єктів може визначити ресторани як основний об'єкт, а Нью-Йорк – як місце розташування, що дозволить пошуковій системі надати релевантні результати, характерні для ресторанів у цьому місці.

Крім того, аналіз настроїв є цінною технікою NLP, яка може покращити пошукові алгоритми, розуміючи настрої або емоційний тон, що стоять за запитами користувачів і пошуковим контентом. Аналіз настроїв передбачає аналіз суб'єктивних думок, ставлень та емоцій, виражених у тексті. Визначаючи сентимент запитів користувачів, пошукові системи можуть адаптувати результати пошуку відповідно до уподобань користувача. Наприклад, якщо користувач шукає «найкращі бюджетні смартфони», аналіз настроїв може виявити позитивний настрій, пов'язаний із запитом, що дозволить пошуковій системі визначити пріоритетність результатів пошуку, які висвітлюють доступні та якісні смартфони.

Методи обробки природної мови також дозволяють пошуковим системам обробляти складніші пошукові запити. Розуміння природної мови, в тому числі діалогових запитів або запитів з декількома компонентами, вимагає розширених можливостей NLP. Пошукові алгоритми можуть використовувати такі методи, як синтаксичний розбір, розбір залежностей і розпізнавання іменованих сутностей для аналізу граматичної структури і залежностей запитів користувачів. Таке глибше розуміння структури запиту дозволяє пошуковим системам точно інтерпретувати наміри користувача і видавати більш релевантні результати пошуку.

Щоб досягти ефективного контекстного розуміння і можливостей НЛП, пошукові системи можуть використовувати попередньо навчені мовні моделі, такі як BERT (Bidirectional Encoder Representations from Transformers) або GPT (Generative Pre-trained Transformer - генеративний попередньо навчений трансформатор). Ці моделі, навчені на величезних обсягах текстових даних, вчаться вловлювати контекстну інформацію, зв'язки між словами і навіть генерувати відповіді, схожі на людські. Включаючи ці мовні моделі в пошукові алгоритми, пошукові системи можуть покращити розуміння запитів користувачів і надавати більш контекстуально релевантні результати пошуку.

4. **Моделі глибокого навчання,** включаючи згорткові нейронні мережі (CNN) і рекурентні нейронні мережі (RNN), стали потужними інструментами в різних задачах обробки природної мови (NLP). Ці моделі можна ефективно використовувати для оптимізації пошукових алгоритмів шляхом вилучення значущих репрезентацій з текстового та візуального контенту. Методи глибокого навчання пропонують значний потенціал для підвищення точності та релевантності результатів пошуку за допомогою таких завдань, як розпізнавання зображень, узагальнення тексту та контекстне розуміння мультимедійного контенту.

Однією з областей, де моделі глибокого навчання досягають успіху, є розпізнавання зображень. Згорткові нейронні мережі (CNN) зробили революцію в галузі комп'ютерного зору завдяки ефективному навчанню та вилученню ієрархічних ознак із зображень. Використовуючи ШНМ, пошукові алгоритми можуть покращити пошук на основі зображень, точно розпізнаючи і класифікуючи візуальний контент. Наприклад, коли користувач надсилає запит на зображення, пошуковий алгоритм може використовувати ШНМ для аналізу зображення запиту і витягувати візуально схожі зображення з пошукового індексу. Таке застосування глибокого навчання для розпізнавання зображень значно покращує пошуковий досвід, дозволяючи користувачам ефективніше знаходити візуально пов'язаний контент.

Узагальнення тексту – ще одна сфера, де моделі глибокого навчання можуть сприяти оптимізації пошукових алгоритмів. ШНМ, особливо такі варіанти, як мережі з довгою короткочасною пам'яттю (LSTM), показали багатообіцяючі результати у створенні стислих і зв'язних анотацій тексту. Навчаючи ШНМ на великих масивах текстових даних, пошукові алгоритми можуть автоматично генерувати резюме веб-сторінок або документів. Ці резюме надають користувачам швидкий огляд контенту, що дозволяє їм ефективно оцінювати релевантність. Включення методів узагальнення тексту, заснованих на моделях глибокого навчання, може вдосконалити пошукові алгоритми, підвищивши ефективність і результативність пошуку інформації.

Контекстне розуміння мультимедійного контенту також є важливим аспектом оптимізації пошукових алгоритмів. Моделі глибокого навчання, такі як мультимодальні мережі злиття, можуть інтегрувати інформацію з різних модальностей, включаючи текст, зображення та аудіо, щоб отримати всебічне розуміння мультимедійного контенту. Одночасно аналізуючи текстові та візуальні підказки, пошукові алгоритми можуть краще розуміти контекст і семантику мультимедійного контенту, що призводить до більш точних і контекстуально релевантних результатів пошуку. Наприклад, коли користувач шукає "найкращі туристичні маршрути", моделі глибокого навчання можуть аналізувати як текстові описи, так і пов'язані з ними зображення, щоб надати повноцінний набір результатів пошуку, які відповідають намірам користувача.

Крім того, моделі глибокого навчання можна використовувати для покращення розуміння та генерації природної мови. Трансформаторні моделі, такі як BERT (Bidirectional Encoder Representations from Transformers) і GPT (Generative Pre-trained Transformer), продемонстрували виняткову ефективність у широкому спектрі завдань NLP. Ці моделі можуть вивчати контекстне представлення слів і речень, що дозволяє пошуковим алгоритмам краще розуміти запити користувачів і зіставляти їх з релевантними результатами пошуку. Використовуючи моделі глибокого навчання для розуміння природної мови, пошукові алгоритми можуть ефективно обробляти складні та нюансовані запити, підвищуючи точність і релевантність результатів пошуку.

5. **Впровадження методів спільної фільтрації та персоналізованих рекомендацій** має величезний потенціал для покращення користувацького досвіду. Аналізуючи поведінку користувачів, їхні вподобання та відгуки, пошукові системи можуть генерувати персоналізовані рекомендації та пропонувати релевантний контент, пристосований до конкретних інтересів користувача. Методи спільної фільтрації також дозволяють пошуковим системам використовувати колективний інтелект і надавати рекомендації на основі вподобань схожих користувачів, що призводить до більш точних і персоналізованих результатів пошуку.

Колаборативна фільтрація – це метод, який використовує мудрість натовпу для створення рекомендацій. Він працює на основі припущення, що користувачі зі схожими смаками та вподобаннями, ймовірно, мають інтереси, які перетинаються. Вивчаючи історичну поведінку та вподобання користувача, алгоритми спільної фільтрації можуть виявити інших користувачів зі схожими шаблонами і рекомендувати елементи або контент, які ці користувачі вважають цінними. Такий підхід дозволяє пошуковим системам використовувати колективний інтелект і фіксувати неявні знання, закладені в спільноті користувачів.

Одним із популярних методів спільної фільтрації є фільтрація на основі користувачів, яка визначає користувачів зі схожими вподобаннями і генерує рекомендації на основі їхніх спільних інтересів. Наприклад, якщо користувач А віддав перевагу певному набору книг або статей, а користувач Б має схожі інтереси, пошукова система може порекомендувати ці матеріали користувачеві Б, припускаючи, що вони можуть його зацікавити. Спільна фільтрація на основі вподобань користувачів може бути особливо ефективною в ситуаціях, коли вподобання користувачів чітко визначені і явно виражені, наприклад, у системах рейтингів або явного зворотного зв'язку.

Інший підхід до спільної фільтрації – це фільтрація на основі об'єктів, яка фокусується на схожості між об'єктами, а не користувачами. Аналізуючи взаємозв'язок між різними об'єктами та їхню спільну появу у взаємодіях або відгуках користувачів, пошукові системи можуть рекомендувати об'єкти, схожі на ті, до яких користувач вже виявляв інтерес. Цей підхід особливо корисний, коли явні вподобання користувача обмежені або недоступні, і дозволяє надавати рекомендації на основі характеристик і властивостей самих об'єктів.

Окрім спільної фільтрації, персоналізовані рекомендації відіграють вирішальну роль в оптимізації пошукових алгоритмів. Персоналізація передбачає адаптацію результатів пошуку та рекомендацій відповідно до вподобань, характеристик та історичної поведінки користувача. Аналізуючи історію пошуку користувача, кількість кліків, попередні взаємодії та явні відгуки, пошукові системи можуть генерувати рекомендації, які відповідають конкретним інтересам і потребам користувача.

Методи персоналізації можуть використовувати алгоритми машинного навчання для створення профілів користувачів та визначення індивідуальних вподобань. Аналізуючи шаблони і тенденції у взаємодії користувача, ці алгоритми можуть визначати відповідні особливості і створювати персоналізовану модель вподобань користувача. Ця модель потім може бути використана для створення рекомендацій, які з більшою ймовірністю будуть резонувати з користувачем. Наприклад, якщо користувач часто шукає і взаємодіє зі статтями на певну тему, пошукова система може розставити пріоритети і порекомендувати схожі статті, щоб покращити користувачеві досвід перегляду.

Більше того, поєднання спільної фільтрації з персоналізованими рекомендаціями може ще більше підвищити точність та ефективність пошукових алгоритмів. Інтегруючи інформацію, отриману як від колективного інтелекту, так і від індивідуальних уподобань користувачів, пошукові системи можуть надавати рекомендації, які враховують як ширші інтереси користувацької спільноти, так і конкретні уподобання окремого користувача. Такий гібридний підхід може запропонувати більш комплексний і персоналізований пошук, гарантуючи, що користувачі отримають релевантний контент, який відповідає їхнім інтересам, а також скористаються мудрістю подібних користувачів.

6. **Інтеграція знань, специфічних для домену**, стає цінним підходом для підвищення точності та релевантності результатів пошуку, особливо в спеціалізованих доменах. Завдяки включенню структурованих даних, онтологій або графів знань, пов'язаних з конкретними доменами, пошукові системи можуть покращити своє розуміння спеціалізованих запитів і надавати адаптовані результати, які відповідають специфічним вимогам домену. Така інтеграція знань про конкретні домени дозволяє пошуковим системам пропонувати точніші та контекстуально відповідні результати пошуку користувачам, які працюють у спеціалізованих доменах.

Одним із важливих аспектів інтеграції знань у певній галузі є використання структурованих даних. Структуровані дані забезпечують формальне представлення інформації з чітко визначеними атрибутами і зв'язками, що дозволяє пошуковим системам витягувати і розуміти дані в систематизованому вигляді. Використовуючи такі формати структурованих даних, як Schema.org або RDF (Resource Description Framework), пошукові системи можуть покращити своє розуміння контенту конкретних доменів і сприяти більш точному узгодженню результатів пошуку. Наприклад, у сфері електронної комерції структуровані дані можуть допомогти пошуковим системам ідентифікувати атрибути продукту, такі як ціна, наявність та оцінки клієнтів, що дозволить користувачам ефективніше знаходити та порівнювати продукти.

Онтології, які представляють знання у формальний та ієрархічний спосіб, є ще одним потужним інструментом для інтеграції знань про предметну область в пошукові алгоритми. Онтології визначають поняття, зв'язки та властивості в межах певної галузі, що дозволяє пошуковим системам фіксувати семантику та знання, пов'язані із запитами користувачів та контентом. Включаючи онтології в пошукові алгоритми, пошукові системи можуть краще розуміти контекст і значення запитів, що призводить до більш точного ранжування результатів пошуку та рекомендацій. Наприклад, у медичній галузі онтологія може визначати взаємозв'язки між хворобами, симптомами і методами лікування, що дозволяє пошуковим системам надавати вичерпну і специфічну для галузі інформацію, коли користувачі шукають інформацію про медичні стани.

Крім того, графи знань забезпечують структуроване представлення знань, з'єднуючи сутності та їхні зв'язки в графоподібну структуру. Створюючи специфічні для домену графи знань, пошукові системи можуть фіксувати взаємозв'язок понять і сутностей у певній галузі. Графи знань полегшують контекстне розуміння і дозволяють пошуковим системам відстежувати взаємозв'язки та надавати більш глибокі та релевантні результати пошуку. Наприклад, у сфері фінансів граф знань може відображати взаємозв'язки між фінансовими інструментами, компаніями та ринковими тенденціями, що дозволяє пошуковим системам надавати вичерпну фінансову інформацію та аналіз.

Інтеграція знань про конкретні домени в пошукові алгоритми дозволяє пошуковим системам долати труднощі, пов'язані з неоднозначними або спеціалізованими запитами. У спеціалізованих доменах користувачі часто використовують термінологію, жаргон або специфічні фрази, які можуть бути незрозумілі широкому загалу. Використовуючи знання про домен, пошукові системи можуть зіставляти запити користувачів з поняттями, характерними для домену, і надавати релевантні результати пошуку навіть тоді, коли точна термінологія може не збігатися. Це підвищує точність результатів пошуку і покращує користувацький досвід для осіб, які працюють у спеціалізованих доменах.

На додаток до підвищення точності результатів пошуку, інтеграція знань про домен може також сприяти використанню додаткових функцій, таких як фасетний пошук і розширені можливості фільтрації. Структуруючи і категоризуючи інформацію про домен, пошукові системи можуть запропонувати користувачам можливість уточнити результати пошуку на основі конкретних атрибутів або аспектів, що мають відношення до домену. Наприклад, у сфері нерухомості користувачі можуть захотіти звузити результати пошуку на основі таких критеріїв, як місце розташування, ціновий діапазон або тип нерухомості. Використовуючи знання про конкретні домени, пошукові системи можуть надавати більш адаптований і настроюваний пошук, який відповідає специфічним потребам користувачів, що працюють у спеціалізованих доменах.

7. **Постійне оцінювання та експерименти** відіграють ключову роль у забезпеченні ефективності алгоритмічних удосконалень і методів ранжування. Щоб досягти цього, розробники пошукових систем повинні створити суворі рамки оцінки, які точно вимірюють ефективність цих удосконалень. Проводячи A/B-тестування, дослідження користувачів та аналіз зворотного зв'язку, можна отримати цінну інформацію про ефективність зусиль з оптимізації. Ця інформація дозволяє розробникам пошукових систем ітеративно вдосконалювати пошукові алгоритми і методи ранжування, що призводить до покращення результатів пошуку і підвищення задоволеності користувачів.

Ключовим аспектом безперервного оцінювання є використання A/B-тестування. A/B-тестування передбачає представлення різних версій пошукового алгоритму або методу ранжування випадково розділеним групам користувачів і вимірювання відповідних показників ефективності. Порівнюючи результати різних алгоритмічних підходів, розробники можуть оцінити вплив конкретних змін на ефективність пошуку. Цей метод дозволяє проводити контрольовані експерименти, де варіації алгоритмічних параметрів, моделей ранжування або наборів функцій можуть бути протестовані в реальних умовах. A/B-тестування надає кількісні докази ефективності алгоритмічних поліпшень і допомагає визначити підходи, які призводять до кращих результатів пошуку.

На додаток до A/B-тестування, дослідження користувачів дають цінну інформацію про користувацький досвід і можуть сприяти оцінці оптимізації пошукових алгоритмів. Користувацькі дослідження передбачають збір якісних відгуків і спостережень за допомогою інтерв'ю, опитувань або аналізу поведінки користувачів. Безпосередньо взаємодіючи з користувачами і фіксуючи їхні думки, розробники пошукових систем можуть отримати глибше розуміння того, як алгоритмічні зміни впливають на задоволеність користувачів, завдання пошуку інформації та загальний досвід пошуку. Дослідження користувачів можуть виявити проблеми юзабіліті, оцінити вподобання користувачів і розкрити аспекти, які не можуть бути відображені лише кількісними показниками. Цей якісний зворотний зв'язок має вирішальне значення для доопрацювання пошукових алгоритмів і забезпечення їхньої відповідності очікуванням і потребам користувачів.

Крім того, аналіз відгуків забезпечує додатковий шлях для безперервного оцінювання. Збираючи та аналізуючи відгуки користувачів, розробники пошукових систем можуть отримати уявлення про сильні та слабкі сторони пошукової системи та визначити сфери для вдосконалення. Відгуки користувачів можна отримати через різні канали, такі як форми зворотного зв'язку, заявки на підтримку або онлайн-форуми. Аналіз цих відгуків може виявити закономірності, тенденції та загальні больові точки, що допоможе розробникам вирішити проблеми користувачів і покращити роботу пошукового алгоритму. Аналіз зворотного зв'язку також сприяє виявленню нових потреб користувачів і тенденцій розвитку пошуку, що дозволяє пошуковим системам адаптуватися і відповідати вимогам користувачів.

Ітеративне вдосконалення пошукових алгоритмів і методів ранжування на основі безперервного оцінювання та експериментів має вирішальне значення для підтримки ефективності та швидкості реагування пошукової системи. Враховуючи відгуки користувачів та інформацію, отриману в результаті A/B-тестування і досліджень користувачів, розробники можуть приймати обґрунтовані рішення щодо вдосконалення алгоритмів, визначати пріоритети оптимізації та усувати недоліки або непередбачувані наслідки. Цей ітеративний процес сприяє створенню середовища безперервного навчання і вдосконалення, гарантуючи, що пошукові алгоритми розвиваються відповідно до мінливих очікувань користувачів і нових проблем пошуку.

Щоб полегшити ефективне безперервне оцінювання та експерименти, розробники пошукових систем повинні створити надійні системи оцінювання, які охоплюють широкий спектр відповідних показників. Ці показники можуть включати показники релевантності (наприклад, точність, згадування), показники залучення користувачів (наприклад, кількість кліків, час перебування на сайті) і показники задоволеності користувачів (наприклад, рейтинги, опитування). Система оцінювання має бути розроблена таким чином, щоб фіксувати як короткострокові покращення продуктивності, так і довгострокову задоволеність користувачів, враховуючи різноманітні потреби та вподобання користувацької бази.

Отже, постійне оцінювання та експерименти мають вирішальне значення для оптимізації пошукових алгоритмів в електронних пошукових сервісах. A/B-тестування, дослідження користувачів та аналіз зворотного зв'язку дають змогу розробникам пошукових систем вимірювати ефективність алгоритмічних удосконалень, збирати інформацію про користувачів та ітеративно вдосконалювати пошукові алгоритми. Інтегруючи кількісні та якісні підходи до оцінки, пошукові системи можуть забезпечити відповідність алгоритмічних удосконалень очікуванням користувачів, покращити ефективність пошуку та загальний користувацький досвід. Майбутні дослідження в цій галузі мають бути зосереджені на розробці передових методологій оцінювання, використанні зворотного зв'язку з користувачами в режимі реального часу та впровадженні методів машинного навчання для автоматизації процесу оцінювання.

## 3.2. Рекомендації щодо майбутніх досліджень і розробок

На додаток до стратегій оптимізації, розглянутих у розділі 3.1, існує кілька ключових сфер, які потребують подальших досліджень і розробок, спрямованих на оптимізацію існуючих технологій для розвитку електронних пошукових сервісів. Ці рекомендації зосереджені на сферах, які доповнюють стратегії, викладені в розділі 3.1, і спрямовані на вирішення існуючих проблем, використання нових технологій і підвищення загальної ефективності та результативності електронних пошукових послуг.

1. **Графи знань** стали потужним інструментом для покращення контекстного розуміння та релевантності в пошукових алгоритмах. Однак існує ще значний потенціал для подальших досліджень і розробок у розширенні та інтеграції графів знань в електронні пошукові сервіси. У цьому розділі представлено рекомендації для майбутніх досліджень у цій галузі.

Одним з ключових аспектів, який слід розглянути, є розширення графів знань для покращення їхньої повноти та охоплення різних сфер. Сучасні графи знань часто спираються на структуровані джерела даних, такі як бази даних і онтології, але існує потреба у включенні знань з різноманітних і неструктурованих джерел. Майбутні дослідження повинні зосередитися на розробці автоматизованих методів вилучення знань з широкого кола джерел, включаючи наукові статті, соціальні мережі, онлайн-форуми і контент, створений користувачами. Ці методи можуть включати обробку природної мови, вилучення інформації та методи машинного навчання для вилучення релевантної інформації та наповнення графа знань. Розширюючи обсяг і глибину графів знань, пошукові системи можуть отримати більш повне розуміння різних доменів, що дозволяє їм надавати більш точні і всеосяжні результати пошуку.

Іншим важливим напрямком досліджень є інтеграція декількох графів знань з різних галузей. Хоча окремі графи знань можуть ефективно охоплювати специфічну для домену інформацію, інтеграція знань з різних джерел може покращити контекстне розуміння та релевантність результатів пошуку. Така інтеграція дозволяє пошуковим системам використовувати міждоменні знання і надавати користувачам більш цілісний досвід пошуку. Дослідницькі зусилля мають бути зосереджені на розробці методів вирівнювання та об'єднання графів знань з різних доменів, долаючи такі виклики, як семантична неоднорідність та відображення схем. Інтегруючи різноманітні графи знань, пошукові системи можуть краще розуміти складні запити, які охоплюють кілька доменів, і виявляти зв'язки та ідеї, які можуть бути неочевидними в межах одного графа знань.

Крім того, у дослідженні слід вивчити методи підвищення точності та свіжості графів знань. Оскільки інформація розвивається і оновлюється з часом, дуже важливо підтримувати графіки знань в актуальному стані. Дослідницькі зусилля мають бути зосереджені на розробці методів безперервного вдосконалення та динамічного оновлення графів знань. Це передбачає моніторинг змін у джерелах даних, визначення відповідних оновлень і своєчасне включення їх у граф знань. Завдяки підтримці точних і актуальних графів знань пошукові системи можуть надавати користувачам найбільш релевантні та своєчасні результати пошуку.

Крім того, дослідження повинні вивчати використання графів знань за межами традиційного текстового пошуку. Графи знань можна інтегрувати з мультимедійним контентом, таким як зображення, відео та аудіо, щоб покращити розуміння контенту та його пошук у мультимодальних сценаріях пошуку. Можна розробити методи, що поєднують візуальну і текстову інформацію, для вилучення значущих зв'язків і відносин між різними модальностями. Така інтеграція графів знань з мультимодальним контентом може дозволити пошуковим системам надавати більш точні та релевантні результати пошуку в сферах, де мультимедіа відіграють значну роль.

2. У контексті цифрових пошукових сервісів, коли пошукові алгоритми стають дедалі складнішими і покладаються на передові технології, такі як машинне та глибинне навчання, потреба в **зрозумілості та прозорості** набуває вирішального значення. Користувачі повинні розуміти, чому їм надаються ті чи інші результати пошуку, і бути впевненими в чесності та неупередженості пошукового процесу. У цьому розділі викладено рекомендації для майбутніх досліджень, спрямованих на підвищення зрозумілості та прозорості пошукових алгоритмів, особливо в сценаріях, де застосовуються моделі глибокого навчання.

Одним із ключових аспектів майбутніх досліджень у цій галузі є розробка методів зрозумілості, які можуть генерувати зрозумілі пояснення рішень щодо ранжування. Моделі глибокого навчання часто працюють як складні чорні ящики, що ускладнює розуміння користувачами основних факторів, які впливають на результати пошуку. Дослідницькі зусилля мають бути зосереджені на розробці методів вилучення інтерпретованої інформації з цих моделей, що дасть змогу генерувати пояснення, які висвітлюють ключові фактори, що впливають на пошукове ранжування. Ці пояснення можуть бути представлені користувачам разом з результатами пошуку, надаючи їм уявлення про те, як обробляються і ранжуються їхні запити.

Крім того, дослідження повинні вивчати методи забезпечення прозорості щодо джерел даних і етапів обробки, які використовуються для формування результатів пошуку. Користувачі повинні мати уявлення про походження та якість даних, що використовуються для формування пошукового рейтингу. Для забезпечення такої прозорості можна дослідити такі методи, як відстеження походження даних та анотування метаданих. Наприклад, пошукові системи можуть надавати інформацію про джерела використаних даних, свіжість даних і будь-які кроки попередньої обробки. Завдяки такій прозорості користувачі можуть оцінити надійність і релевантність представленої їм інформації, що дозволить їм робити більш обґрунтовані висновки.

Окрім зрозумілості та прозорості для користувачів, дослідження також повинні бути спрямовані на розробку методів аудиту та перевірки справедливості та неупередженості пошукових алгоритмів. Упередженість і дискримінація можуть ненавмисно проникати в пошукові алгоритми, впливаючи на результати пошуку та користувацький досвід. Дослідницькі зусилля мають бути спрямовані на вивчення методів виявлення та пом'якшення упередженості пошукових алгоритмів, особливо тих, що виникають через навчальні дані або алгоритмічні упередження. Це включає розробку підходів до виявлення упереджених шаблонів у результатах пошуку, оцінку справедливості алгоритмів ранжування для різних демографічних груп і впровадження механізмів для усунення та виправлення будь-яких виявлених упереджень.

Також слід докласти зусиль для розробки стандартів і рекомендацій щодо зрозумілості та прозорості пошукових алгоритмів. Ці стандарти можуть допомогти розробникам пошукових систем впроваджувати найкращі практики надання пояснень, розкриття джерел даних і забезпечення справедливості та неупередженості. Крім того, дослідження повинні зосередитися на оцінці та порівнянні різних методів пояснення та механізмів прозорості, щоб визначити їхні сильні сторони, обмеження та потенційні компроміси.

3. Зростаюча поширеність мультимедійного контенту в цифрових пошукових сервісах вимагає зосередити майбутні дослідження на розробці методів ефективного пошуку та розуміння мультимодального контенту, зокрема зображень, відео та аудіо.

Щоб уможливити ефективний **мультимодальний пошук**, дослідницькі зусилля мають бути зосереджені на розробці алгоритмів і моделей, які можуть аналізувати та інтерпретувати візуальні та аудіальні аспекти контенту. Сюди входить розвиток таких технологій, як розпізнавання зображень, розуміння відео та обробка аудіо. Наприклад, алгоритми розпізнавання зображень можуть бути розроблені для виокремлення релевантних візуальних ознак та ідентифікації об'єктів, сцен і патернів на зображеннях. Аналогічно, методи розуміння відео можуть уможливити аналіз відеоконтенту, включаючи відстеження об'єктів, розпізнавання дій і виявлення подій. Крім того, досягнення в обробці аудіо, такі як перетворення мови в текст і класифікація аудіо, можуть полегшити розуміння аудіоконтенту і дозволити пошуковим системам ефективно обробляти усні запити і аудіоінформацію.

На додаток до аналізу окремих модальностей, алгоритми мультимодального пошуку повинні бути спрямовані на вилучення та використання інформації з різних модальностей для кращого розуміння запитів користувачів і доступного контенту. Це вимагає розробки методів інтеграції та об'єднання інформації з різних модальностей. Наприклад, поєднання текстового аналізу з візуальними і слуховими характеристиками може призвести до більш повного розуміння контенту і підвищити точність результатів пошуку. Дослідницькі зусилля мають бути зосереджені на розробці методів мультимодального злиття, які ефективно поєднують інформацію з різних модальностей для покращення розуміння контенту та його пошуку.

Крім того, алгоритми мультимодального пошуку повинні враховувати контекстні зв'язки між різними модальностями. Наприклад, розуміння зв'язку між зображенням і пов'язаним з ним текстовим або звуковим описом може сприяти більш точній інтерпретації контенту. Такі методи, як крос-модальний пошук і вирівнювання, можуть бути використані для встановлення значущих зв'язків між різними модальностями і отримання більш точних результатів пошуку.

Щоб полегшити мультимодальний пошук, дослідження повинні також вивчати розробку користувацьких інтерфейсів і парадигм взаємодії, які підтримують введення і пошук мультимодальних запитів. Користувацькі інтерфейси повинні дозволяти користувачам вводити запити різними способами, наприклад, текстом, голосом або зображенням, а також надавати інтуїтивно зрозумілі способи взаємодії з результатами мультимодального пошуку та їхнього вивчення. Це вимагає розробки зручних інтерфейсів, які підтримують різні способи введення і представляють результати пошуку в зрозумілій і візуально привабливій формі.

4. Крім того, дослідження повинні бути зосереджені на вирішенні проблем **масштабованості та ефективності,** пов'язаних з мультимодальним пошуком. Оскільки обсяг і складність мультимодального контенту продовжують зростати, вкрай важливо розробити ефективні механізми індексації та пошуку, здатні обробляти великомасштабні мультимодальні дані. Це передбачає оптимізацію методів зберігання і обробки, а також розробку алгоритмів, які можуть ефективно отримувати і ранжувати результати мультимодального пошуку в режимі реального часу.

5. Оскільки електронні пошукові сервіси продовжують розвиватися і збирають велику кількість даних про користувачів, вкрай важливо враховувати **етичні міркування** і надавати пріоритет захисту приватності користувачів. Майбутні дослідження мають бути зосереджені на розробці пошукових алгоритмів, що зберігають конфіденційність і забезпечують баланс між персоналізованим пошуком і захистом приватності користувачів. Такі методи, як федеративне навчання, диференційована конфіденційність і безпечні багатосторонні обчислення, можуть бути досліджені, щоб забезпечити захист даних користувачів і водночас уможливити персоналізований пошук.

Одним із ключових аспектів майбутніх досліджень у цій галузі є розробка методів збереження конфіденційності, які дозволять пошуковим системам використовувати дані користувачів, зберігаючи при цьому їхню приватність. Федеративне навчання, наприклад, дозволяє навчати моделі на децентралізованих джерелах даних без необхідності передавати конфіденційні дані користувачів на центральний сервер. Диференційовані методи конфіденційності можуть бути використані для додавання шуму або випадковості до даних перед аналізом, гарантуючи, що індивідуальна інформація користувача залишається приватною, але при цьому дозволяє проводити агрегований аналіз для покращення пошукових алгоритмів. Безпечні методи багатосторонніх обчислень можуть уможливити спільний аналіз і обмін інформацією без розкриття вихідних даних, захищаючи конфіденційність окремих користувачів.

6. Окрім захисту приватності, надзвичайно важливим є **усунення упередженості** пошукових алгоритмів. Дослідницькі зусилля мають бути зосереджені на розробці методів виявлення та пом'якшення упередженості пошукових алгоритмів, оскільки упередженість може призвести до несправедливих і дискримінаційних результатів. Сюди відносяться упередженість у результатах пошуку, упередженість при ранжируванні та демографічна упередженість. Вкрай важливо розробити справедливі та неупереджені моделі ранжування, які справедливо ставляться до всіх користувачів і контенту. Для усунення упереджень і забезпечення справедливого представлення інформації в результатах пошуку можна використовувати такі методи, як попередня обробка даних, алгоритмічні коригування та ранжування з урахуванням розмаїття.

Крім того, етичні міркування пошукових алгоритмів виходять за межі приватності та упередженості. Пошукові системи зобов'язані просувати достовірну та надійну інформацію, боротися з дезінформацією та забезпечувати прозорість своїх джерел даних і методологій ранжування. Майбутні дослідження повинні зосередитися на розробці методів виявлення та зменшення поширення неправдивої інформації та дезінформації в результатах пошуку. Це включає розробку методів перевірки контенту, оцінки достовірності джерел і коригування алгоритмів для зменшення видимості оманливої або неправдивої інформації. Підвищення прозорості пошукових алгоритмів може передбачати розкриття джерел використаних даних, критеріїв ранжування та надання пояснень до представлених результатів пошуку.

Крім того, дослідницькі зусилля повинні бути спрямовані на розробку механізмів контролю та згоди користувачів. Користувачі повинні мати можливість розуміти і контролювати збір, використання та поширення своїх даних у пошукових службах. Орієнтовані на користувача налаштування конфіденційності, механізми інформованої згоди та інтуїтивно зрозумілі засоби контролю конфіденційності можуть дати користувачам можливість приймати обґрунтовані рішення щодо своїх даних та уподобань у сфері конфіденційності.

Насамкінець, майбутні дослідження повинні визначити пріоритетом розробку пошукових алгоритмів, що зберігають конфіденційність і забезпечують баланс між персоналізованим пошуком і захистом приватності користувачів. Усунення упередженості пошукових алгоритмів має важливе значення для забезпечення справедливості та рівноправного представництва. Крім того, слід зосередити зусилля на просуванні достовірної інформації, боротьбі з дезінформацією та підвищенні прозорості пошукових алгоритмів. Розвиток механізмів контролю та згоди користувачів може розширити їхні можливості контролювати свої дані та вподобання щодо конфіденційності. Звертаючись до цих напрямків досліджень, електронні пошукові сервіси можуть підтримувати етичні принципи, захищати конфіденційність користувачів і сприяти цілісності та достовірності результатів пошуку.

7. Зі збільшенням обсягу та швидкості поширення інформації в Інтернеті зростає попит на можливості динамічного пошуку в режимі реального часу. Пріоритетом майбутніх досліджень має стати розробка методів, які дозволять пошуковим системам обробляти і видавати результати пошуку в режимі реального часу, забезпечуючи актуальний і своєчасний пошук інформації. У цьому розділі представлено рекомендації щодо розвитку динамічного пошуку в реальному часі для задоволення потреб і очікувань користувачів, які постійно змінюються.

Ефективні механізми індексування та пошуку є ключовими для забезпечення пошуку в режимі реального часу. Традиційні підходи до індексування можуть не впоратися з масштабом і швидкістю, з якою генерується нова інформація. Майбутні дослідження мають бути зосереджені на розробці методів індексування, які можуть обробляти великі потоки даних і швидко включати новий контент до пошукового індексу. Це передбачає вивчення структур даних і алгоритмів, які оптимізують операції зберігання, пошуку та оновлення для сценаріїв пошуку в реальному часі. Крім того, дослідницькі зусилля мають бути спрямовані на підвищення ефективності обробки запитів, щоб уможливити швидке отримання результатів пошуку з оновленого індексу.

Динамічний пошук вимагає пошукових алгоритмів, які можуть адаптуватися до динаміки інформації та вподобань користувачів. Персоналізація відіграє вирішальну роль у забезпеченні динамічного пошуку. Майбутні дослідження повинні зосередитися на включенні контексту і вподобань користувача в пошукові алгоритми для надання персоналізованих і динамічних результатів пошуку. Контекстно-орієнтовані методи пошуку можуть враховувати такі фактори, як місцезнаходження користувача, час і пристрій, щоб забезпечити більш релевантні та персоналізовані результати пошуку. Адаптивні системи рекомендацій можуть використовувати моделі поведінки користувачів, зворотній зв'язок і попередню історію пошуку, щоб динамічно коригувати ранжування і представлення результатів пошуку. Враховуючи контекст і вподобання користувача, пошукові системи можуть видавати не лише актуальні результати пошуку, але й такі, що відповідають конкретним потребам та інтересам окремих користувачів.

Для підтримки динамічного пошуку в реальному часі дослідницькі зусилля також повинні бути спрямовані на розробку ефективної і масштабованої інфраструктури. Сюди входять фреймворки та архітектури розподілених обчислень, які можуть обробляти та аналізувати великі обсяги даних у режимі реального часу. Такі технології, як потокова обробка, паралельні обчислення і розподілене кешування, можуть бути використані для створення ефективних і масштабованих систем пошуку в реальному часі. Крім того, інтеграція технологій хмарних і периферійних обчислень може підвищити оперативність і зменшити затримку пошукових сервісів, що дозволить швидше знаходити і надавати результати пошуку.

Крім того, оцінка систем пошуку в реальному часі та динамічного пошуку має важливе значення для забезпечення їхньої ефективності та зручності використання. У майбутніх дослідженнях слід розробити систему оцінювання та метрики, які відображатимуть продуктивність і зручність пошуку в режимі реального часу. Дослідження користувачів, аналіз відгуків і порівняльні оцінки можуть дати цінну інформацію про ефективність і результативність алгоритмів і систем пошуку в реальному часі. Безперервне оцінювання та експерименти мають вирішальне значення для ітеративного вдосконалення та оптимізації методів пошуку в реальному часі та динамічного пошуку.

Отже, майбутні дослідження і розробки в галузі електронних пошукових сервісів мають бути зосереджені на розширенні графів знань, забезпеченні зрозумілості і прозорості, розвитку мультимодального пошуку і розуміння контенту, врахуванні етичних міркувань і захисту приватності, а також забезпеченні можливостей пошуку в режимі реального часу і динамічного пошуку. Ці рекомендації спрямовані на подальшу оптимізацію існуючих технологій, вирішення поточних проблем і підвищення ефективності, результативності та зручності користування електронними пошуковими послугами. Розвиваючи дослідження в цих сферах, ми можемо очікувати на значний прогрес в електронних пошукових сервісах, які краще задовольнятимуть зростаючі потреби користувачів у дедалі складнішому цифровому ландшафті.

## Висновки до розділу 3

Ефективні пошукові алгоритми та точні методи ранжування є ключовими для розвитку електронних пошукових сервісів. Для оптимізації існуючих технологій у цій галузі дуже важливо зосередитися на модернізації пошукових алгоритмів і методів ранжування.

Тому майбутні дослідження і розробки в галузі електронних пошукових сервісів повинні бути зосереджені на розширенні графів знань, забезпеченні зрозумілості та прозорості, просуванні мультимодального пошуку і розуміння контенту, врахуванні етичних міркувань і захисту приватності, а також наданні можливостей пошуку в реальному часі і динамічного пошуку. Ці рекомендації спрямовані на подальшу оптимізацію існуючих технологій, вирішення актуальних проблем і підвищення ефективності, результативності та зручності електронних пошукових сервісів. Розвиваючи дослідження в цих сферах, ми можемо очікувати значного прогресу в електронних пошукових сервісах, які краще задовольнятимуть зростаючі потреби користувачів у дедалі складнішому цифровому ландшафті.

# ВИСНОВКИ

Концепція цифрових пошукових сервісів відіграє вирішальну роль у забезпеченні ефективного та дієвого доступу до цифрового контенту. Вони дозволяють користувачам орієнтуватися у величезному цифровому просторі та знаходити потрібну їм інформацію.

Склад та архітектура електронних пошукових сервісів є ключовими компонентами у розробці ефективних та дієвих інформаційно-пошукових систем. Вони забезпечують системну організацію даних, визначають порядок видачі результатів пошуку, обробляють запити та полегшують взаємодію з користувачами.

Технології, що використовуються в електронних пошукових сервісах, постійно розвиваються, що призводить до покращення пошукового досвіду користувачів та розширення їхніх можливостей у доступі до цифрової інформації. Однак електронні пошукові сервіси також стикаються з такими викликами, як оцінка якості та надійності джерел інформації, фільтрація неправдивої інформації та забезпечення конфіденційності користувачів. Подолання цих викликів вимагає міждисциплінарних досліджень та розробки інноваційних рішень. Пошук інформації, обробка природної мови, етика даних, взаємодія людини з комп'ютером та інформатика є важливими сферами для вирішення цих питань. Таким чином, загальна концепція передбачає постійне вдосконалення цифрових пошукових сервісів за допомогою технологічних і наукових досягнень для забезпечення ефективного і зручного доступу до цифрової інформації.

Тому порівняння характеристик базових моделей і алгоритмів пошуку дозволяє дослідникам і розробникам приймати обґрунтовані рішення при виборі та впровадженні відповідних технологій для електронних пошукових сервісів, виходячи з їхніх конкретних вимог. Такі фактори, як виразність, стандартизація, оптимізація запитів і зручність для користувача, є важливими міркуваннями при виборі мови пошуку інформації. Специфічні вимоги пошукового сервісу, характер запитуваних даних і цільова аудиторія користувачів повинні бути враховані при прийнятті обґрунтованих рішень.

Щодо методів ранжування, то їхня основна мета – ранжування найбільш релевантних результатів. Однак релевантність – це складне поняття. Ми не маємо чіткого розуміння того, що є релевантним документом для конкретного запиту, і це поняття стає ще більш невловимим, коли ми шукаємо релевантні документи, які відповідають цілям інформаційного пошуку користувача. Між вісцеральною потребою користувача (нечітким сприйняттям інформаційної проблеми у свідомості користувача) і компромісною потребою (формулюванням запиту в рамках обмежень наявного пошукового інструменту) відбуваються численні трансформації. Порівнюючи і розуміючи характеристики цих базових моделей і алгоритмів пошуку, дослідники і розробники можуть приймати обґрунтовані рішення при виборі і впровадженні відповідних технологій для електронних пошукових сервісів, виходячи з їхніх конкретних вимог. Такі фактори, як виразність, стандартизація, оптимізація запитів і зручність для користувача, є важливими міркуваннями при виборі мови пошуку інформації.

Тому загальна концепція передбачає постійне вдосконалення цифрових пошукових сервісів на основі технологічного та наукового прогресу для забезпечення ефективного та зручного доступу до цифрової інформації. Вирішуючи ці проблеми та розвиваючи дослідження в цих сферах, ми можемо очікувати значного прогресу в електронних пошукових сервісах, які краще задовольнятимуть зростаючі потреби користувачів у дедалі складнішому цифровому ландшафті.

# СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. А Б. Д. Оптимізація модуля пошуку даних в інформаційних системах : thesis. 2009. URL: http://er.nau.edu.ua/handle/NAU/14957 (дата звернення: 23.05.2023).
2. Зв'язування даних. Дія.Відкриті дані. URL: https://diia.data.gov.ua/info-center/linkeddata (дата звернення: 23.05.2023).
3. О Т. Огляд існуючих алгоритмів та моделей пошуку у Web. URL: http://masters.donntu.ru/2013/fknt/arbuzova/library/article8.htm.
4. Основні моделі пошуку. URL: https://ua-referat.com/uploaded/osnovni-modeli-poshuku/index1.html (дата звернення: 23.05.2023).
5. С К. Г., Сергіївна К. Г. Методи порівняльного аналізу сучасних пошукових систем : thesis. 2008. URL: http://er.nau.edu.ua/handle/NAU/14949 (дата звернення: 23.05.2023).
6. Учасники проектів Вікімедіа. DPLL алгоритм. Вікіпедія. URL: https://uk.wikipedia.org/wiki/DPLL\_алгоритм (дата звернення: 23.05.2023).
7. Учасники проектів Вікімедіа. SQL – вікіпідручник. Вікіпідручник. URL: https://uk.wikibooks.org/wiki/SQL (дата звернення: 23.05.2023).
8. Учасники проектів Вікімедіа. Векторна модель. Вікіпедія. URL: https://uk.wikipedia.org/wiki/Векторна\_модель (дата звернення: 23.05.2023).
9. Baeza-Yates R. Modern information retrieval. New York : ACM Press, 1999. 513 p.
10. Brin S., Page L. The anatomy of a large-scale hypertextual Web search engine. Computer networks and ISDN systems. 1998. Vol. 30, no. 1-7. P. 107–117. URL: https://doi.org/10.1016/s0169-7552(98)00110-x (date of access: 23.05.2023).
11. H M. J., Jurafsky D. Speech and language processing: pearson new international edition. Pearson Education, Limited, 2013. 944 p.
12. Harman D. Interactive evaluation. Information retrieval evaluation. Cham, 2011. P. 57–75. URL: https://doi.org/10.1007/978-3-031-02276-0\_3 (date of access: 23.05.2023).
13. Haveliwala T. H. Topic-sensitive pagerank: a context-sensitive ranking algorithm for web search. IEEE transactions on knowledge and data engineering. 2003. Vol. 15, no. 4. P. 784–796. URL: https://doi.org/10.1109/tkde.2003.1208999 (date of access: 23.05.2023).
14. Interpreting in multilingual, multicultural contexts. Choice reviews online. 2010. Vol. 48, no. 04. P. 48–1893–48–1893. URL: https://doi.org/10.5860/choice.48-1893 (date of access: 23.05.2023).
15. Introduction to information retrieval. Choice reviews online. 2009. Vol. 46, no. 05. P. 46–2715–46–2715. URL: https://doi.org/10.5860/choice.46-2715 (date of access: 23.05.2023).
16. Introduction to information retrieval. The Stanford Natural Language Processing Group. URL: https://nlp.stanford.edu/IR-book/information-retrieval-book.html (date of access: 23.05.2023).
17. Järvelin K., Kekäläinen J. Cumulated gain-based evaluation of IR techniques. ACM transactions on information systems. 2002. Vol. 20, no. 4. P. 422–446. URL: https://doi.org/10.1145/582415.582418 (date of access: 23.05.2023).
18. Kutsenko O., Serdiuk O. Peculiarities of work of plagiate search algorithms in software. Cherkasy university bulletin: applied mathematics. informatics. 2020. No. 1. P. 75–85. URL: https://doi.org/10.31651/2076-5886-2019-1-75-85 (date of access: 23.05.2023).
19. Learn cypher in Y Minutes. Scenic Programming Language Tours. URL: https://learnxinyminutes.com/docs/uk-ua/cypher-ua/ (date of access: 23.05.2023).
20. Learning to rank by optimizing expected reciprocal rank / P. Zhang et al. Information retrieval technology. Berlin, Heidelberg, 2011. P. 93–102. URL: https://doi.org/10.1007/978-3-642-25631-8\_9 (date of access: 23.05.2023).
21. Ma N., Guan J., Zhao Y. Bringing PageRank to the citation analysis. Information processing & management. 2008. Vol. 44, no. 2. P. 800–810. URL: https://doi.org/10.1016/j.ipm.2007.06.006 (date of access: 23.05.2023).
22. Meta-search and multi-domain search / S. Ceri et al. Web information retrieval. Berlin, Heidelberg, 2013. P. 161–179. URL: https://doi.org/10.1007/978-3-642-39314-3\_11 (date of access: 23.05.2023).
23. Metaxas P. T., Mustafaraj E. Social media and the elections. Science. 2012. Vol. 338, no. 6106. P. 472–473. URL: https://doi.org/10.1126/science.1230456 (date of access: 23.05.2023).
24. Polynczuk-Alenius K. Algorithms of oppression: how search engines reinforce racism. Information, communication & society. 2019. Vol. 22, no. 13. P. 2030–2032. URL: https://doi.org/10.1080/1369118x.2019.1593485 (date of access: 23.05.2023).
25. Query DSL | elasticsearch guide [8.7] | elastic. Elasticsearch Platform. URL: https://www.elastic.co/guide/en/elasticsearch/reference/current/query-dsl.html (date of access: 23.05.2023).
26. Robertson S., Zaragoza H. Probabilistic relevance framework: BM25 and beyond. Now Publishers, 2009. 70 p.
27. Salton G. Introduction to modern information retrieval. New York : McGraw-Hill, 1983. 448 p.
28. Sivic J., Zisserman A. Efficient visual search of videos cast as text retrieval. IEEE transactions on pattern analysis and machine intelligence. 2009. Vol. 31, no. 4. P. 591–606. URL: https://doi.org/10.1109/tpami.2008.111 (date of access: 23.05.2023).
29. Welcome to apache lucene. Apache Lucene. URL: https://lucene.apache.org/ (date of access: 23.05.2023).
30. Witten I. H., Moffat A., Bell T. C. Managing gigabytes: compressing and indexing documents and images. IEEE transactions on information theory. 1995. Vol. 41, no. 6. P. 2101. URL: https://doi.org/10.1109/tit.1995.476344 (date of access: 23.05.2023).
31. XQuery tutorial. W3Schools Online Web Tutorials. URL: https://www.w3schools.com/xml/xquery\_intro.asp (date of access: 23.05.2023).

# ДОДАТКИ

**Додаток А. Список запитів для тестування методів ранжування**

|  |  |
| --- | --- |
| **Ідентифікатор запиту** | **Запит** |
| 1 | Штучний інтелект |
| 2 | Зміна клімату |
| 3 | Інтелектуальний аналіз даних |
| 4 | Відновлювана енергетика |
| 5 | Алгоритми машинного навчання |
| 6 | Генна інженерія |
| 7 | Квантові обчислення |
| 8 | Стале сільське господарство |
| 9 | Кібербезпека |
| 10 | Обробка природної мови |
| 11 | Робототехніка |
| 12 | Інтернет речей |
| 13 | Аналітика великих даних |
| 14 | Біомедичні дослідження |
| 15 | Доповнена реальність |
| 16 | Віртуальна реальність |
| 17 | Технологія блокчейн |
| 18 | Нанотехнології |
| 19 | Бездротовий зв'язок |
| 20 | Хмарні обчислення |
| 21 | Конфіденційність в Інтернеті |
| 22 | Соціальні мережі |
| 23 | Штучні нейронні мережі |
| 24 | Криптовалюта |
| 25 | Електронна комерція |
| 26 | Комп'ютерний зір |
| 27 | Мобільні додатки |
| 28 | Цензура в Інтернеті |
| 29 | Автономні транспортні засоби |
| 30 | Взаємодія людини та комп'ютера |
| 31 | Ігрова індустрія |
| 32 | Інформатика в охороні здоров'я |
| 33 | Квантова криптографія |
| 34 | Кібер-фізичні системи |
| 35 | Прогнозування стихійних лих |
| 36 | Інтернет медичних речей |
| 37 | Алгоритмічна торгівля |
| 38 | Архітектури нейронних мереж |
| 39 | 3D-друк |
| 40 | Робототехніка в охороні здоров'я |
| 41 | Етика штучного інтелекту |
| 42 | Хмарні рішення для зберігання даних |
| 43 | Інтернет транспортних засобів |
| 44 | Технології відновлюваної енергетики |
| 45 | Майнінг криптовалют |
| 46 | Положення про конфіденційність даних |
| 47 | Біометрична автентифікація |
| 48 | Додатки для машинного навчання |
| 49 | Аналітика соціальних мереж |
| 50 | Загрози кібербезпеці |
| 51 | Безпека Інтернету речей |
| 52 | Граничні обчислення |
| 53 | Додатки для інтернету речей |
| 54 | Штучний інтелект в охороні здоров'я |
| 55 | Автоматизація розумного будинку |
| 56 | Технологія безпілотників |
| 57 | Візуалізація даних |
| 58 | Квантове машинне навчання |
| 59 | Датчики Інтернету речей |
| 60 | Генетичні алгоритми |
| 61 | Автономні дрони |
| 62 | Роботизована автоматизація процесів |
| 63 | Віртуальні помічники |
| 64 | Штучний інтелект у фінансах |
| 65 | Платформи Інтернету речей |
| 66 | Блокчейн у ланцюжку поставок |
| 67 | Генерація природної мови |
| 68 | Охорона здоров'я за допомогою Інтернету речей |
| 69 | Предиктивна аналітика |
| 70 | Технологія розпізнавання обличчя |
| 71 | Штучний інтелект у маркетингу |
| 72 | Машинне навчання у фінансах |
| 73 | Інтернет речей у сільському господарстві |
| 74 | Рамки кібербезпеки |
| 75 | Стандарти Інтернету речей |
| 76 | Робототехніка на виробництві |
| 77 | Штучний інтелект в освіті |
| 78 | Машинне навчання в охороні здоров'я |
| 79 | Підключення до Інтернету речей |
| 80 | Найкращі практики кібербезпеки |
| 81 | Алгоритми квантових обчислень |
| 82 | Енергія Інтернету речей |
| 83 | Технологія чат-ботів |
| 84 | Штучний інтелект у логістиці |
| 85 | Машинне навчання в ритейлі |
| 86 | Інтернет речей на транспорті |
| 87 | Управління ризиками кібербезпеки |
| 88 | Додатки для глибокого навчання |
| 89 | Інтернет речей у розумних містах |
| 90 | Аналіз настроїв |
| 91 | Штучний інтелект в обслуговуванні клієнтів |
| 92 | Алгоритми машинного навчання у фінансах |
| 93 | Інтернет речей в охороні здоров'я |
| 94 | Реагування на інциденти кібербезпеки |
| 95 | Розуміння природної мови |
| 96 | Штучний інтелект на виробництві |
| 97 | Машинне навчання в ланцюжку поставок |
| 98 | Конфіденційність даних Інтернету речей |
| 99 | Поінформованість про кібербезпеку |
| 100 | Алгоритми розпізнавання зображень |

**Додаток Б. Реалізація функції ранжування BM25 на Python**

import math

class BM25:

def \_\_init\_\_(self, corpus, tokenizer=None):

self.corpus\_size = len(corpus)

self.avgdl = sum(len(doc) for doc in corpus) / self.corpus\_size

self.doc\_freqs = []

self.idf = {}

self.doc\_len = []

self.tokenizer = tokenizer if tokenizer else lambda x: x.split()

self.\_initialize(corpus)

def \_initialize(self, corpus):

for doc in corpus:

frequencies = {}

for token in self.tokenizer(doc):

if token not in frequencies:

frequencies[token] = 0

frequencies[token] += 1

self.doc\_len.append(sum(frequencies.values()))

self.doc\_freqs.append(frequencies)

for token, freq in frequencies.items():

if token not in self.idf:

self.idf[token] = 0

self.idf[token] += 1

for token, freq in self.idf.items():

self.idf[token] = math.log(self.corpus\_size - freq + 0.5) - math.log(freq + 0.5)

def get\_scores(self, query):

scores = [0] \* self.corpus\_size

query\_tokens = self.tokenizer(query)

for token in query\_tokens:

if token not in self.idf:

continue

for i in range(self.corpus\_size):

doc\_freq = self.doc\_freqs[i].get(token, 0)

numerator = self.idf[token] \* doc\_freq \* (2.2 + 1)

denominator = doc\_freq + 1.2 \* (1 - 0.75 + 0.75 \* self.doc\_len[i] / self.avgdl)

scores[i] += numerator / denominator

return scores

**Додаток В. Пошуковий алгоритм, розроблений з використанням rank\_bm25**

from rank\_bm25 import BM25Okapi

# Define the corpus of documents

corpus = [

"This is the first document",

"This is the second document",

"This is the third document",

"This is the fourth document",

"This is the fifth document"

]

# Tokenize the corpus

tokenized\_corpus = [doc.split(" ") for doc in corpus]

# Create a BM25Okapi object

bm25 = BM25Okapi(tokenized\_corpus)

# Define the query

query = "first document"

# Tokenize the query

tokenized\_query = query.split(" ")

# Get the BM25 scores for each document

doc\_scores = bm25.get\_scores(tokenized\_query)

# Sort the documents by their scores

sorted\_docs = sorted(zip(doc\_scores, corpus), reverse=True)

# Print the top 3 documents

for score, doc in sorted\_docs[:3]:

print(f"Score: {score:.4f} \t Document: {doc}")

**Додаток Г. Модифікація, яка дозволяє читати пошуковій системи читати PDF-файли**

import math

import glob

import PyPDF2

class BM25:

def \_\_init\_\_(self, corpus, tokenizer=None):

self.corpus\_size = len(corpus)

self.avgdl = 0

self.doc\_freqs = []

self.idf = {}

self.doc\_len = []

self.tokenizer = tokenizer if tokenizer else lambda x: x.split()

self.\_initialize(corpus)

def \_initialize(self, corpus):

for doc\_path in corpus:

with open(doc\_path, 'rb') as file:

pdf = PyPDF2.PdfFileReader(file)

doc = ''

for page in range(pdf.getNumPages()):

doc += pdf.getPage(page).extract\_text()

frequencies = {}

for token in self.tokenizer(doc):

if token not in frequencies:

frequencies[token] = 0

frequencies[token] += 1

self.doc\_len.append(sum(frequencies.values()))

self.doc\_freqs.append(frequencies)

for token, freq in frequencies.items():

if token not in self.idf:

self.idf[token] = 0

self.idf[token] += 1

self.avgdl = sum(self.doc\_len) / self.corpus\_size

for token, freq in self.idf.items():

self.idf[token] = math.log(self.corpus\_size - freq + 0.5) - math.log(freq + 0.5)

def get\_scores(self, query):

scores = [0] \* self.corpus\_size

query\_tokens = self.tokenizer(query)

for token in query\_tokens:

if token not in self.idf:

continue

for i in range(self.corpus\_size):

doc\_freq = self.doc\_freqs[i].get(token, 0)

numerator = self.idf[token] \* doc\_freq \* (2.2 + 1)

denominator = doc\_freq + 1.2 \* (1 - 0.75 + 0.75 \* self.doc\_len[i] / self.avgdl)

scores[i] += numerator / denominator

return scores

# Example usage

corpus\_files = glob.glob('path/to/corpus/\*.pdf')

bm25 = BM25(corpus\_files)

query = "example query"

scores = bm25.get\_scores(query)

print(scores)