МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ТЕХНОЛОГІЙ ТА ДИЗАЙНУ
Факультет ринкових, інформаційних та інноваційних технологій
Кафедра інформаційно-комп'ютерних технологій
та фундаментальних дисциплін

Дипломна магістерська робота

на тему:

«ОПТИМІЗАЦІЯ СИСТЕМИ ПОШУКУ ІНФОРМАЦІЇ В МЕРЕЖІ ІНТЕРНЕТ»

Виконала: студентка групи МгЧКІ-20
спеціальності 123 Комп'ютерна інженерія
освітньої програми Комп'ютерна інженерія
Анастасія ОКАПІНСЬКА

Керівник: к.е.н., доцент
Наталія БАБІНА
Рецензент: _____
(прізвище та ініціали)

Черкаси 2021

ЗМІСТ

ВСТУП

Актуальність теми. Проблема пошуку інформації в мережі стає усе більш актуальною останнім часом. Основні протоколи, використовувані в Internet, не забезпечені достатніми вбудованими функціями пошуку, не говорячи вже про мільйони серверів, що перебувають у ній. Протокол HTTP, використовуваний в Internet, гарний лише відносно навігації, яка розглядається тільки як засіб перегляду сторінок, але не їхнього пошуку. Те ж саме стосується й протоколу FTP, який навіть більш примітивний, ніж HTTP. Через швидкий ріст інформації, доступної в Мережі, навігаційні методи перегляду швидко досягають межі їх функціональних можливостей, не говорячи вже про межу їх ефективності. Не вказуючи конкретних цифр, можна сказати, що потрібну інформацію вже не представляється можливим одержати відразу, тому що в Мережі зараз перебувають мільярди документів і всі вони в розпорядженні користувачів Internet, до того ж сьогодні їх кількість зростає згідно з експонентною залежністю. Збільшення загального числа документів в WWW у цілому погіршило картину доступності інформації. Кількість змін, яким ця інформація піддана, величезне й, найголовніше , вони відбулися за дуже короткий період часу. Основна проблема полягає в тому, що єдиної повної функціональної системи відновлення й занесення подібного обсягу інформації, одночасно доступного всім користувачам Internet в усьому світі, ніколи не було. Для того, щоб структурувати інформацію, накопичену в мережі Internet, і забезпечити її користувачів зручними засобами пошуку необхідних їм даних, були створені пошукові системи. Але й вони не можуть дати стовідсотковий результат того, що прагне знайти користувач.

Ріст обсягів даних, які необхідно розміщати, передавати, а головне знаходити, привів до того, що системи зберігання даних, засновані на класичній клієнт-серверній архітектурі, практично досяглися межі своєї масштабованості. Це спровокувала поява й усе більш широка поширення децентралізованих мереж. Децентралізовані мережі забезпечували фантастичний рівень масштабованості, однак з ростом таких мереж виникли проблеми пошуку в них

(пошуку вузлів, ресурсів, інформації). Виниклі проблеми були викликані децентралізованим характером таких мереж і, як наслідок, відсутністю єдиного координаційного центру.

Будь-яка проблема рано або пізно вирішується. І в цьому випадку теж з'являються спеціалізовані алгоритми пошуку, оптимізовані для використання мережі, вони сфокусовані на володіння низьким навантаженням на мережу й порівняно невеликим відсотком неодружених (повторних) сканувань вузлів, простоту у реалізації. Але потрібно відзначити, що завдання підвищення точності пошуку в Інтернеті не завжди добре вирішується чисто математичними методами.

Таким чином, розробка інформаційно-пошукової системи з використанням нових методів оптимізації пошуку інформації в мережі Internet є вельми актуальною.

Мета і завдання дослідження. Метою дослідження є оптимізація пошукової системи для більш точного та простого здійснення пошуку.

Для досягнення цієї мети в дипломній роботі розв’язані такі задачі:

- Проаналізовано сучасні підходи до оптимізації пошукових систем;
- Визначено найефективнішу математичну модель ранжування результатів;
- Розроблено алгоритм для реалізації програмної частини пошукового комплексу;
- Реалізована структура програмного застосування;
- Наведена інструкція з пошуку;

Об’єкт дослідження – процес створення пошукової системи в глобальній мережі Інтернет.

Предмет дослідження – інформаційні технології розробки пошукової системі.

Методи дослідження. Для визначення структури оптимізації пошукової системи на основі лінгвістичного процесору використані методи застосування лінгвістичних функцій; для визначення алгоритмічної організації

перефразовування – методи лексичного аналізу параметричних слів; для програмної реалізації та експериментальної перевірки розроблених методів – методи моделювання з використанням у якості програмного інтерпретатора гнучкої, невибагливої до системних ресурсів скриптової мови.

Наукова новизна одержаних результатів:

– отримав подальший розвиток метод лінгвістичної оптимізації для інформаційно-пошукової системи, який дає можливості спрощення побудови пошукового запиту для користувача;

– визначена найоптимальніша математична модель ранжування результатів, згідно з якою був розроблений засіб для подальшої реалізації програмної частини комплексу, модулі індексування сторінок і ранжирування результатів пошуку, а також допоміжний модуль включений до комплексу, який веде статистики по кількості входжень слів на web-сторінках і запитів, які забезпечують більш точну видачу результатів пошукової системи;

– запропоновано застосування метапошукової структури для збільшення якості отриманих результатів, при використанні лінгвістичного методу оптимізації;

– розроблено програмне забезпечення реалізації пошукової системи, яке дозволяє спростити процес пошуку інформації в мережі Інтернет для користувачів.

Практичне значення одержаних результатів полягає у вдосконаленні інформаційно-пошукової системи для більш зручного й простого використання користувачами.

На основі визначення можливостей лінгвістичного аналізатора, перспектива перифразування лежить у розширенні числа використовуваних лексичних функцій і більш точного їхнього настроювання на різні типи інформації., що дозволяє розширювати можливості чіткого пошуку різного типу інформації.

Розроблено програмний продукт (з використанням гнучкої скриптової мови), що дозволяє покращити та прискорити пошукові можливості користувача.

Апробація результатів магістерської роботи. Основні положення магістерського дослідження доповідались та обговорювались на XVII студентській науковій конференції «Інформаційні інновації сучасного українського суспільства», напрям Інформаційно-комп'ютерний, жовтень 2021 р.

Публікації. За результатами магістерської роботи було опубліковано статтю, яка відображає основні результати роботи.

РОЗДІЛ 1
ПОСТАНОВКА І АНАЛІЗ ЗАДАЧІ

Завданням дипломної магістерської роботи є дослідження можливості оптимізації алгоритму пошуку інформації в Інтернеті.

Для здійснення можливості оптимізації дослідити запропонований метод і реалізацію його застосування. Для аналізу результатів використовувати пошуковий комплекс.

Комплекс повинен складатися з таких програмних модулів:

- crawler («павука»), який переміщається по мережі й збирає інформацію;
- лінгвістичний процесор для перефразувань;
- пошуковий механізм, використовуваний як інтерфейс.

Основні вимоги до системи:

- підтримка HTTP, HTTP - Проксі, HTTPS;
- розпізнавання з наступним індексуванням:
  - ✓ класичних посилань виду <a href..>;
  - ✓ посилань із документів, що містять фрейми;
  - ✓ сценаріїв клієнтської мови JavaScript;
- підтримка text/html, text/xml, text/plain, та image/gif;
- індексування мультиязычных сайтів.
- підтримка <META NAME="robots" content="..."> и robots.txt.

Для формування видачі результатів пошуковою системою вибрати найбільш оптимальну математичну модель ранжирування.

Система повинна формувати статистичні таблиці по ключових словах web- ресурсів з індексу.

РОЗДІЛ 2
ОГЛЯД АНАЛОГІВ ПОШУКОВИХ СИСТЕМ

2.1 Порівняльні характеристики

На сьогодні у світі існує досить велика кількість систем, які спеціалізуються на пошуку інформації в Internet. Я розгляну лише деякі, самі популярні системи, представлені в російськомовному сегменті Internet. Це Meta і Google.

Meta - найбільший український пошуковий ресурс. Пошукова система Meta дозволяє шукати по всьому українському Internet, а також по реєстру українських сайтів. Повнотекстовий пошук іде з обліком росіянці й української морфології. У багатослівних запитах система не ігнорує так звані "стоп-слова", до яких відносять приводи, частки, союзи й т.п. Більшість пошукових систем при пошуку їх ігнорують, тобто , при запиті крем від засмаги привід "від" буде зігнорований і серед результатів будуть документи зі словосполученням "крем для засмаги". видасть документи, які точно збігаються із запитом.

Google – сама популярна пошукова система у світі. Google переведений на 26 мов, може знаходити інформацію на 186 мовах, найбільші портали в усьому світі вибрали Google у якості пошукового інструмента (з недавніх пор компанія Mail.ru стала використовувати технологію пошуку Google). Новаторська технологія Pagerank дозволила створити якісний пошук і видавати при запиті більш релевантні результати.

Користувачі Google можуть знайти інформацію на різних мовах, перевірити котирування акцій, знайти карти, що течуть новини, звістки пошук серед 1 мільярда картинок, а також скористатися найбільшим архівом повідомлень у світі Usenet - більш мільярда повідомлень, відправлених з 1981 року. Зручність і простота Google зробили його одним з найвідоміших брэндов у світі, інформація про який поширювалася в основному від одного задоволеного користувача до іншого. Систима продовжує удосконалитися й останнім часом одержала багато нововведень, таких як пошук по зображеннях

Image Swirl, пошук по кодах Google Code Search і остання розробка Google Squared - це пошукова система Google, яка видає дані у вигляді таблиці, з коротким описом кожного ресурсу, який надає інформацію із запиту [1].

Таблиця 1.1 – Основні характеристики пошукових систем

Назва ПС	Google	Meta
Адрес	google.com	meta.ua
Кіл-ть унікальних користувачів у день (середнє за тиждень із 13.05.10 по 21.05.10)	23 607 903	224 342
Розмір пошукової системи (на 21.05.10)	26 530 000 000	124 582 318
Глибина індексування	не обмежена	не обмежена
Підтримка фреймів	+	+
Підтримка ImageMaps	+	+
Індексація закритих розділів	+	+
Показники популяр-ності web-сайта	+	-
Визначення частоти відновлення	-	-
Robots.txt	+	+
Можливість перевірки сторінки на наявність в індексах	+	+
Meta Robots	+	+

Таблиця 1.2 – Фактори, що впливають на визначення релевантності сторінки

Назва ПС	Google	Meta
Адреса	google.com	meta.ua
Мета-теги	+	+
Індексація в поле ALT	+	+
Наявність зовнішніх посилань	+	+
Морфологічний пошук	+	+
Облік регістру	+	-
Невидимий текст	SPAM	SPAM
Зайва повторюваність ключових слів	SPAM	SPAM
Дрібний текст	SPAM	SPAM

Розмір пошукової системи - загальна кількість проіндексованих сторінок. Від розміру пошукової системи залежить, чи буде Web-Сайт

представлений у її індексах, скільки сторінок Web-Сайту буде проіндексовано і т.д.

Глибина індексування показує, скільки сторінок крім зазначеної буде індексувати пошукова система. Як правило, у великих пошукових машин немає обмеження на глибину, і їх роботи намагаються проіндексувати усі сторінки Web-Сайту. Це не завжди виходить, тому що на їхньому шляху можуть виникнути перешкоди, наприклад, такі як фрейми, Image maps, динамічно створені сторінки і т.д.. Ряд пошукових систем (наприклад, Infoseek, Lycos) при індексації обмежуються лише деякою кількістю сторінок Web-Сайту.

Підтримка фреймів. Деякі пошукові системи не розуміють фреймової структури сайту. Внаслідок цього практично всі сторінки Вашого сайту можуть бути не проіндексовані.

Підтримка Imagemaps. Не всі пошукові системи можуть випливати по посиланнях, зазначених за допомогою image maps.

Індексація закритих розділів. Ряд пошукових машин можуть індексувати захищені розділи на серверах, якщо їм указати login і пароль. Користувач не зможе відразу перейти на захищену сторінку й вивчити всю інформацію, але завдяки пошуковій системі він буде знати, що така інформація існує й, можливо, прийме розв'язок заплатити й одержати до неї доступ.

Показники популярності Web-Сайту. Пошукова система може визначити "популярність" Web-Сайту по кількості посилань на нього з інших Web-Ресурсів. "Популярність" може бути одним з факторів у прийнятті системою розв'язку про те, індексувати даний Web-Сайт чи ні. Для Яндексу існує показник ТИЦ - тематичний індекс цитування, для Google це показник PR (Pagerank), ці показники свідчать про зовнішні посилання на Web-Сайт.

Визначення частоти відновлення. Деякі пошукові машини визначають, наскільки часто обновляються ті або інші сторінки. Дана інформація допомагає відповідним чином спланувати графік повторних візитів роботів для переіндексації сторінок. Часто обновлювані ресурси відвідуються частіше, статичні сторінки - рідше.

Robots.txt, Meta Robots. У силу деяких обставин адміністратор сайту може не бажати індексації всіх або певних сторінок свого Web-Ресурсу. Уникнути індексації можна двома шляхами:

- за допомогою файла Robots.txt, розміщеного на Web-сервері.

- за допомогою спеціального мета-тегу, який міститься на конкретну

сторінку Web-Сайту й пропонує роботам не затягати її в індекси системи. Виглядає в такий спосіб:

<META NAME="ROBOTS" CONTENT="NOINDEX">

Можливість перевірки сторінки на наявність в індексах. Дуже корисна опція, якої мають далеко не всі пошукові машини. Дозволяє визначити наявність в індексах системи тієї або іншої сторінки й подивитися, як вона виглядає в системі. Для людини, що займається просуванням Web-Ресурсу, немаловажне знати, які ресурси мережі містять на нього посилання, у якому контексті це посилання використовується і т.д. Тому можливість висновку сайтів, що містять подібні посилання, надає пошуковій системі додаткову цінність.

Мета-теги. Не всі системи підтримують позначка-теги: description і keywords, тобто враховують ключові слова, що втримуються в цих тегах.

Індексація в поле ALT. Не всі системи враховують ключові слова, що втримуються в поле ALT тегу IMG, при визначенні релевантності сторінки. ля довідки: у поле ALT заноситься альтернативний текстовий підпис до картинок на сторінці.

Стоп-слова. Для економії місця й збільшення продуктивності деякі пошукові системи не включають в індекси слова, що зустрічаються на Web-Сторінках дуже велике кіл- у раз. Наприклад, "www", артикли "a", "the" і т.д.

Морфологічний пошук. Якщо пошукова система підтримує морфологію, то пошук буде здійснюватися не тільки по зазначенім слову, але й по всіх його морфологічних формах. Тобто, наприклад, при запиті "банер" така пошукова машина знайде також сторінки, що містять "банера ", "банерів" і т.д.

Облік регістру. Деякі пошукові системи чутливі до запитів з урахуванням регістру, інші - немає. Наприклад, пошукова система Altavista при запиті "banner" видасть Вам усі сторінки, що містять слово "banner", де букви можуть бути в будь-якому регістрі, але при запиті "Banner" - тільки сторінки, що містять це слово із заголовною першою буквою.

Спам пошукових систем. Цілком зрозуміле прагнення кожного Web-Майстра добитися того, щоб при запиті по певних ключових словах його сторінка видавалася якнайближче до початку списку. Іноді бажання добитися успіху на цім поприщі штовхає деяких використовувати непривабливі приймання штучного збільшення релевантності свій сторінки - спамить пошукові системи. В основному, спам полягає у використанні невиправдано великого кількості ключових слів на сторінці. Причому їх намагаються використовувати там, де вони мають найбільшу "вагу" для пошукової системи, - у заголовку сторінки (тегу title), назвах розділів і т.д. Часто для того, щоб подібні додаткові слова не псували відвідувачам враження від сторінки, їх пишуть текстом, що збігаються по кольору із тлом сторінки, пишуть їхнім дрібним шрифтом і т.д.

Зрозуміло, адміністрація розвідувачів не схвалює подібні дії. Системи пошуку покликано знаходити й відображати документи відповідно до того, що містить текст, призначений для відвідувачів, а не по "збагаченій суміші" ключових слів.

Великі міжнародні розвідувачі застосовують ряд заходів щодо боротьби зі спамом. Якщо таким системам попадеться сторінка, яка містить у позначка - тегу keywords те саме слово більш 5 раз, або, наприклад, невидимий для відвідувачів текст, вона не буде проіндексована системою.

2.2 Алгоритм ранжування сторінок у Google. PageRank

Механізм ранжирування - це програма, яка визначає релевантність сторінки (ступінь відповідності) пошуковому запиту на основі семантичного аналізу документа, щільності й відповідності ключових слів, посилань із інших

ресурсів і інших параметрів. Від релевантності сторінки залежить її місце при висновку результатів пошуку.

PageRank – це метод Google для виміру "важливості" сторінки. Коли всі інші фактори, такі як тэг Title і ключові слова враховані, Google використовує Pagerank, щоб відкоригувати результати так, що більш "важливі" сайти піднімуться відповідно нагору на сторінці результатів пошуку користувача [3].

Тобто , порядок ранжирування в Google працює в такий спосіб:

1 Знайти всі сторінки, відповідні до ключових слів пошуку.

2 Відранжувати відповідно "сторінковим факторам", таким, як ключові слова.

3 Урахувати текст посилань на сторінки.

4 Відкоригувати результати даними PageRank.

Якщо Сторінка A посилається на сторінку B, то Сторінка A вважає, що Сторінка B - важлива сторінка. Текст посилання не використовується в PageRank. PageRank також впливає на важливість посилань на сторінку. Якщо на сторінку вказують багато важливих посилань, то її посилання на інші сторінки також стають більш важливими.

Технологія Pagerank працює так: нехай на сторінці Pj розміщене lj посилань. Якщо одна із цих посилань веде на сторінку Pi, то Pj передасть 1/lj своєї "важливості" сторінці Pi.

Рівень важливості (тобто , PR) сторінки Pi є сума всіх таких значень із усіх вхідних посилань. Якщо представити набір сторінок, що посилаються на сторінку Pi, як Bi, то "важливість" Pi обчислюється по наступній формулі:

$$I(P_i) = \sum_{P_j \in B_i} \frac{I(P_j)}{l_j} \qquad (2.1)$$

Щоб довідатися PR сторінки, нам потрібно спочатку знати PR усіх сторінок, які на неї посилаються. Втім, математичні методи дозволяють розв'язати і цю проблему.

Для цього створюється матриця гіперпосилань $H = [H_{ij}]$, у якій рядок і стовпця j буде мати такий вигляд:

$$H_{ij} = \begin{cases} 1/l_j & \text{if } P_j \in B_i \\ 0 & \text{otherwise} \end{cases} \quad (2.2)$$

Це стохастична матриця, тобто матриця, у якій усі стовпці й/або рядка - ряди ненегативних дійсних чисел, що дають у сумі одиницю.

Слід сформувати вектор $I = [I(P_i)]$, елементами якого є значення PR, тобто "важливість" усіх сторінок. За умовою вектор виходить стаціонарним. Розгляду підлягає ситуація на прикладі невеликої матриці з восьми web- сторінок, гіперпосилання між якими відображаються стрілками.

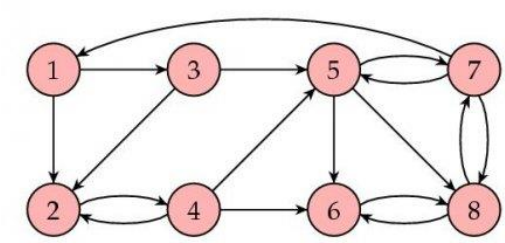

Рисунок 2.1 – Матриця сторінок

Цій ситуації відповідає така матриця:

$$\mathbf{H} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1/3 & 0 \\ 1/2 & 0 & 1/2 & 1/3 & 0 & 0 & 0 & 0 \\ 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/2 & 1/3 & 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 1/3 & 1/3 & 0 & 0 & 1/2 \\ 0 & 0 & 0 & 0 & 1/3 & 0 & 0 & 1/2 \\ 0 & 0 & 0 & 0 & 1/3 & 1 & 1/3 & 0 \end{bmatrix}$$

і стаціонарний вектор

$$I = \begin{bmatrix} 0.0600 \\ 0.0675 \\ 0.0300 \\ 0.0675 \\ 0.0975 \\ 0.2025 \\ 0.1800 \\ 0.2950 \end{bmatrix}$$

Розрахунки показує, що сторінка 8 має найбільша вага. На рис 2.2 показане найбільше "авторитетні" сторінки, і пофарбовані вони більш світлим кольором.

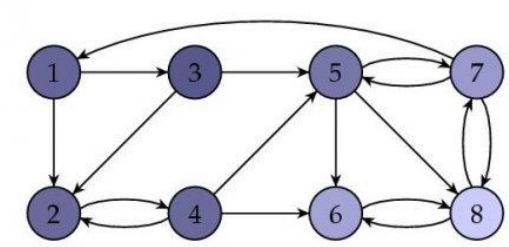

Рисунок 2.2 – Матриця найбільш авторитетних сторінок

Вага PageRank - самий важкий для маніпулювання фактор при оптимізації сторінок. Вага Pagerank важко як одержати, так і удержати.

2.3 Визначення популярності web-ресурсу в Яндекс. ТІЦ

Індекс цитування (або ТІЦ) — прийнята в науковому світі захід "значимості" праць якого-небудь ученого. Величина індексу визначається кількістю посилань на цю працю (або прізвище) в інших джерелах. Однак для дійсно точного визначення значимості наукових праць важливо не тільки кількість посилань на них, але і якість цих посилань. Так, на роботу може посилатися авторитетне академічне видання, популярна брошура або розважальний журнал. Значимість у таких посилань різна. Тематичний індекс цитування (ТІЦ) визначає "авторитетність" web-ресурсів з урахуванням якісної характеристики посилань на них з інших сайтів. Цю якісну характеристику називають "вагою" посилання. Розраховується вона по спеціально розробленому алгоритму. Більшу роль відіграє тематична близькість ресурсу сайтів, що й посилаються на нього. Сама по собі кількість посилань на ресурс також впливає на значення його ТІЦ, але ТІЦ визначається не кількістю посилань, а сумою їх ваг.

ТІЦ як засіб визначення авторитетності ресурсів покликано забезпечити релевантність розташування ресурсів у рубриках каталогу Яндексу. ТІЦ не є

чисто кількісною характеристикою, тому показуються деякі округлені значення, які допомагають орієнтуватися в "значимості" ("авторитетності") ресурсів у кожній області (темі).

ТІЦ розраховується для web- ресурсів. Під web-ресурсом може розумітися як сайт, так і деякий розділ сайту (фізично це директорія). Розділ сайту (директорія) уважається самостійним ресурсом, якщо вона описана в каталозі Яндексу. Якщо для сайту в каталозі описано кілька директорій, ТІЦ буде обраховуватися для кожної з них, а якщо ні, то весь сайт буде вважатися одним web- ресурсом.

Оскільки в ТІЦ ураховується тільки вага зовнішніх web- ресурсів, що посилаються на заданий, ТІЦ не може бути збільшений ні за рахунок "внутрішніх" посилань (з одних сторінок ресурсу на інші), ні за рахунок розташування декількох посилань на одній або декількох сторінках того самого "зовнішнього" ресурсу. При розрахунках ТІЦ одного з розділів сайту (директорій) посилання на розділ сайту з інших розділів цього сайту будуть уважатися внутрішніми й, отже, не будуть збільшувати його ТІЦ. При цьому посилання на кожний з розділів сайту враховуються (поєднуються) при підрахунку ТІЦ усього сайту [2].

2.4 Принципи роботи пошукових систем

2.4.1 Індексування

Практично всі великі пошукові системи мають свою власну структуру, відмінну від інших. Однак можна виділити загальні для всіх пошукових машин основні компоненти. Відмінності в структурі можуть бути лише у вигляді реалізації механізмів взаємодії цих компонентів.

Модуль індексування складається із трьох допоміжних програм (роботів):

Spider (павук) – програма, призначена для завантаження Web-Сторінок. "Павук" забезпечує завантаження сторінки й витягає всі внутрішні посилання із цієї сторінки. Завантажується html- код кожної сторінки. Для завантаження сторінок роботи використовують протоколи HTTP. Працює "павук" у такий

спосіб. Робот на сервер передає запит “get/path/document” і деякі інші команди Http- Запиту. У відповідь робот одержує текстовий потік, що містить службову інформацію й безпосередньо сам документ. Посилання витягуються з тегів a, area, base, frame, frameset, і ін. Поряд з посиланнями, багатьма роботами обробляються редиректи (перенапрямку). Кожна завантажена сторінка зберігається в наступному форматі:

- URL сторінки;
- дата, коли сторінка була завантажена;
- http-заголовок відповіді сервера;
- тіло сторінки (html-код);

Crawler («подорожуючий» павук) – програма, яка автоматично проходить по всіх посиланнях, знайдених на сторінці. Виділяє всі посилання, що присутні на сторінці. Його завдання - визначити, куди далі повинен іти павук, ґрунтуючись на посиланнях або виходячи із заздалегідь заданого списку адрес. Crawler, випливаючи по знайдених посиланнях, здійснює пошук нових документів, ще невідомих пошукової системі.

Indexer (робот-індексатор) - програма, яка аналізує Web-Сторінки, завантаженні павуками. Індексатор розбирає сторінку на складові частини й аналізує їх, застосовуючи власні лексичні й морфологічні алгоритми. Аналізу зазнають різні елементи сторінки, такі як текст, заголовки, посилання структурні й стильові особливості, спеціальні службові html- теги і т.д.

Таким чином, модуль індексування дозволяє обходити по посиланнях задана безліч ресурсів, завантажувати сторінки, що зустрічаються, витягати посилання на нові сторінки з одержуваних документів і робити повний аналіз цих документів.

2.4.2 Ранжування

Робот-індексатор поміщає оброблену сторінку в базу даних. База даних, або індекс пошукової системи - це система зберігання даних, інформаційний

масив, у якім зберігаються спеціальним образом перетворені параметри всіх завантажених і оброблених модулем індексування документів.

Пошуковий сервер є найважливішим елементом усієї системи, тому що від алгоритмів, які лежать в основі її функціонування, прямо залежить якість і швидкість пошуку.

Пошуковий сервер працює в такий спосіб: отриманий від користувача запит зазнає морфологічному аналізу. Генерується інформаційне оточення кожного документа, що втримується в базі (яке й буде згодом відображене у вигляді сниппета, тобто відповідної до запиту текстової інформації на сторінці видачі результатів пошуку).

Отримані дані передаються як вхідні параметри спеціальному модулю ранжирування. Відбувається обробка даних по всіх документах, у результаті чого, для кожного документа розраховується власний рейтинг, що характеризує релевантність запиту, уведеного користувачем, і різних складових цього документа, що зберігаються в індексі пошукової системи.

Залежно від вибору користувача цей рейтинг може бути скоректований додатковими умовами (наприклад, так званий "розширений пошук").

Далі генерується сниппет, тобто , для кожного знайденого документа з таблиці документів витягують заголовок, коротка анотація, найбільш відповідна запиту й посилання на сам документ, причому знайдені слова підсвічуються.

Отримані результати пошуку передаються користувачеві у вигляді SERP (Search Engine Result Page) - сторінки видачі пошукових результатів.

База даних відшукує предмет запиту, заснований на інформації, зазначеної в заповненій формі, і виводить відповідні документи, підготовлені базою даних. Щоб визначити порядок, у якім список документів буде показаний, база даних застосовує алгоритм ранжирування. В ідеальному випадку, документи, найбільш релевантні користувацькому запиту будуть поміщені першими в списку. Різні пошукові системи використовують різні алгоритми ранжирування, однак основні принципи визначення релевантності наступні:

- кількість слів запиту в текстовому вмісті документа (тобто в html-коді);

- текст вхідного посилання;

- використання ключових слів у тегі Title;

- якість і релевантність зовнішніх посилань;

- ключові слова в заголовках H1, h2, h3, h(x), <b>, ;

- ключові слова в мета-тегах Description, Keywords;

- якість контенту документа, вік документа, вік домена;

- місце розташування шуканих слів у документі;

- питома вага слів, щодо яких визначається релевантність, у загальній кількості слів документа.

Негативні фактори, що впливають на ранжирування:

- сервер часто недоступний пошуковому роботові;

- контент дуже схожий або дублює зміст уже наявного в індексі сайту;

- участь у обмінно-посилальних схемах, або активна торгівля посиланнями;

- занадто більша щільність ключових слів, тобто "нудота" сторінки;

- занадто великий час відповіді сервера;

- зовнішні посилання дуже низької якості (спам);

- вхідні посилання з підозрілих на погляд пошукової системи сайтів.

2.5 Огляд алгоритмів роботи пошукових систем

Задаючи однаковий запит у різних розвідувачах ви зверніть увагу, що видача в них відрізняється. Відповідь полягає в тому, що всі пошукові системи поводяться по різному, але основна причина полягає в тому що розвідувачі використовує різні алгоритми. Цей порядок роботи алгоритмів необхідний пошуковим системам для визначення релевантності відповідно до запиту користувача. Алгоритм пошукових систем розглядаються як математичні формули, яка ухвалюється для всіх розв'язків. Алгоритм використовує ключові запити й надає релевантні результати у вигляді розв'язку завдань. Ключові запити визначаються пошуковими роботами, де перевіряється контент сторінки

й релевантність запитів на основі формул алгоритмів, які в кожної пошукової системи різні.

Є сервіси які збирають інформацію про запити, які часто зустрічаються і про сторінки, що найбільше часто проглядаються, і часу, витраченого на кожну сторінку. Отримана інформація застосовується для видачі результатів, які самі популярні в користувачів. Безліч запитів до яких застосована ця технологія спричиняє спам. Ще один підхід ураховує аналіз посилань, де хороші тематичні сторінки посилаються на інші гарні тематичні сторінки. Визначаючи як посилаються ці сторінки один на одного й розвідувач визначає яка сторінка релевантна. Точно так само, деякі алгоритми пошукових систем відображають внутрішню посилальну структуру на малюнку. Дотримуючись внутрішніх посилань для оцінки простоти навігації й оцінки співвідношення сторінок.

Ці бази даних створювалися на основі згрупованій користувачем інформації. Даний метод розглядається як архаїчний, хоча існує не мало директорій, що становлять бази пошукових систем, такі як Open Directory і DMOZ, які групуються вручну. Матеріали в деяких пошукових системах формується вручну, як тільки пошукові роботи зберуть необхідну інформацію. Алгоритми аналізують розташування ключових слів на сторінках з високою частотністю сприймаються як більш релевантні, це називається щільність ключових слів. Узагальнено алгоритм роботи пошукової системи й рейтинг, який вона вишиковує на основі запиту (ключове слово), ураховує й аналізує наступне:

1 Загальна кількість ключових слів на сайті.

2 Загальна кількість ключових слів на сторінці.

3 Співвідношення загального числа слів на сайті до кількості ключових слів на сайті.

4 Співвідношення загального числа слів на сторінці до кількості ключових слів на сторінці.

5 Індекс цитування.

6 Популярність тематики.

7 Число запитів по конкретнім ключовім слову за певний період часу.

8 Загальна кількість сторінок сайту.

9 Застосування стилю до сторінок сайту.

10 Загальний обсяг тексту сайту.

11 Загальний обсяг сайту.

12 Загальний обсяг кожної сторінки сайту.

13 Загальний обсяг тексту кожної сторінки сайту.

14 Вік сайту.

15 Назва URL сайту (ім'я домена)

16 Періодичність відновлення інформації на сайті.

17 Останнє відновлення сторінок сайту.

18 Загальне число картинок (малюнків) на сайті.

19 Загальна кількість мультимедійних файлів.

20 Наявність написів, що заміщають, на малюнках (картинках).

21 Довжину (у кількості символів), що заміщають написів малюнків (картинок).

22 Використання фреймів.

23 Мова сайту (російський або іноземний).

24 Розмір шрифту, яким оформлені ключові слова.

25 Жирність шрифту ключових слів.

26 Написані в розрядку чи ні ключові слова.

27 Написані чи ні заголовними буквами ключові слова.

28 Як далеко від початку сторінки розташовуються ключові слова.

29 Стиль заголовків і найменувань ключових слів.

30 Наявність і аналіз позначка тегів.

31 Наявність і зміст опису й властивостей сторінки.

32 Наявність файлу "робот".

33 Географічне місце розташування сайту.

34 Коментарі усередині програмного коду сайту.

35 ДО якого типу сторінок ставиться кожна сторінка сайту : html або asp.

36 Наявність у складі сайту flash модулів.

37 Наявність у складі сайту сторінок з незначними відмінностями друг від друга.

38 Відповідність ключових слів сайту тому розділу каталогу пошукової машини, у якім зареєстрований сайт.

39 Наявність "шумових слів" ("стоп слів").

40 Загальна кількість гіперпосилань сайту.

41 Кількість внутрішніх гіперпосилань сайту.

42 Кількість зовнішніх гіперпосилань сайту.

43 Глибина сайту.

44 Ряд інших спеціальних технічних параметрів.

2.6 Механізми пошуку

У сфері пошукових технологій, безсумнівно, присутні незаперечні гранди, такі як Google, Yahoo, MSN. Але, крім перерахованих, існують також інші цікаві розв'язки успішного пошуку в мережі Інтернет. Сьогодні новачки пошукового ринку можуть запропонувати користувачеві можливості, яких немає в лідерів. Розглянемо принципи й концепції побудови альтернативних механізмів пошуку.

Механізм - штучний інтелект. Використання штучного інтелекту (Artificial Intelligence) - перспективний розв'язок у багатьох областях. Ціль застосування цієї технології - добитися "людського" розуміння пошукового запиту. За допомогою цього можна добитися більш релевантних результатів (виключення з результату текстів, що підходять тільки по ключових словах, а не за змістом). Ще одна перевага полягає у формуванні запитів на "природній мові", а не набором ключових слів. Одним з лідерів у цій області є пошукова система hakia.com, запущена в експлуатацію (бета-версія) у листопаді 2006 року. Деякі результати даної машини дійсно здатні вразити, однак не всі так гладко, з деякими запитами система не справляється. Розроблювачам має бути прикласти ще багато сил, але перспективність даного напрямку дуже висока.

Механізм - пошук силами користувачів. Пошук силами користувачів означає об'єднання машинних і людських зусиль. Головний принцип - люди шукають краще, чим машини. Одним із прикладів є сервіс del.icio.us, який, по суті, не є пошуковою системою, а представляє посилання з описами, ретельно відібрані користувачами. Дуже цікавим розв'язком є розвідувач chacha.com, який пропонує користувачеві при пошуку потрібної інформації скористатися допомогою працівника сервісу у формі чата. Chacha.com платить своїм " Інтернет-помічникам" близько 5 у.е. у годину, що робить саму організацію пошукового механізму більш витратної.

Механізм - персоналізований пошук. Дана технологія заснована на різних результатах пошуку в різних користувачів. Відмінності ґрунтуються на статистику пошукових фраз даного користувача, на настроювання області пошуку (можна шукати тільки по певному колу сайтів). Приклади впровадження такого виду пошуку: у Росії - сервіс personal.novoteka.ru, у світовому масштабі - сервіс google.com/coop.

Механізм - вистава результатів. Поліпшити якість пошуку можна також, поліпшивши сприйняття його результатів. У цьому напрямку працюють кілька цікавих сервісів. Наприклад пошукова система quintura.com (розроблена в Росії) представляє результати у вигляді хмари тегів, які дозволяють уточнити результат. Пошукова машина kartoo.com використовує кластеризацію пошукових результатів і генерує на цій основі дерево термінів, що ставляться до пошукового запиту користувача.

Механізм - вертикальний пошук. Вертикальний пошук полягає в пошуку тільки по сайтах з певною тематикою. Наприклад, тільки по автомобільних сайтах, туристичних агентствах і т.д. Це дозволяє набагато поліпшити релевантність пошуку. Одним із прикладів даного напрямку є популярний розвідувач по блогам technorati.com або simplyhired.com, що спеціалізується на пошуку по сайтах із пропозиціями про роботу й резюме. [4]

2.7 Пошукова оптимізація

Пошукова оптимізація - це комплекс робіт над сайтом і зовнішніми факторами для досягнення найкращих позицій у пошукових системах відповідно до обраних ключових слів.

Пошукову оптимізацію можна розділити на внутрішню й зовнішню.

Внутрішня оптимізація сайту спрямована на роботу із самим сайтом. До неї ставиться:

1 Складання семантичного ядра сайту.

Семантичне ядро являє собою сукупність запитів (ключових слів), змісту яких відповідає ресурс. Семантичне ядро створюється з обліком специфіки сайту з найпоширеніших і відповідних ключових слів. За таким списком ключових слів відслідковується просування сайту.

2 Оптимізація сторінок сайту.

У неї входять роботи з html- кодом і текстами (контентом) сторінок. При оптимізації html- коду проводиться виправлення безпосередньо html- коду, корекція Meta- Тегів, заголовків, описів сторінок сайту, виділення потрібних частин сторінки спеціальними тегами. Усі тексти сторінок аналізуються й коректуються відповідно до ключових слів.

3 Оптимізація структури сайту.

Зміна внутрішніх посилань на сторінки, створення карти сайту, для того щоб пошуковий робот зміг проіндексувати усі сторінки. Після таких робіт пошуковим роботам буде простіше й зручніше працювати зі сторінками, що прискорить їхню індексацію.

До зовнішньої оптимізації ставляться дії по підвищенню "дружності" до пошукових систем і авторитетності (популярності) ресурсу. Щоб збільшити популярність сайту потрібно врахувати такі фактори як:

1 Посилання із сайтів з більшим ТІЦ і PageRank.

Такі посилання є якісними й мають більшу вагу, що впливає на позиції сайту в результатах пошуку.

2 Доречні тексти опису посилань.

Текст посилання, що містить ключові слова, сприймається пошуковою системою як додаткова рекомендація, що підтверджує відповідність пошуковому запиту, що впливає на ранжирування сайту.

3 Посилання на тематичних сайтах.

Крім тексту посилань пошукові роботи враховують загальний інформаційний уміст сторінки, що посилається, сайту й при схожості тематик дають таким посиланням більша вага.

4 Однобічні посилання.

Пошукові системи намагаються відслідковувати взаємні посилання, тому віддають перевагу однобічним посиланням, уважаючи їх більш справжніми й коштовними.

З тих пір як збільшення посилань стала одним з важливих факторів ранжирування, число сайтів "каталогів посилань" зросло. Пошукові системи негативно ставляться до численних каталогів сайтів і намагаються знецінити такі посилання або не враховувати їх зовсім.

2.8 Математичні моделі інформаційного пошуку

У даний момент часу спостерігається ріст кількість різних джерел інформації, доступної в мережі Internet. Усе більше значення набуває проблема пошуку потрібних даних з величезного числа документів. Часто мова йде вже не про те, щоб знайти книгу або статтю, де згадується інформація, що цікавить людину, а у виборі із сотень і тисяч подібних джерел найбільш підходяще потрібне питання, що й повно освячує, документ.

Майже 60% пошукових систем і модулів функціонують без усяких математичних моделей, а при пошуку у великому обсязі інформації, потоці користувацьких запитів, крім емпірично поставлених коефіцієнтів, корисним виявляється оперувати теоретичним апаратом, нехай і не складним.

Модель пошуку - це комбінація наступних складових [7, с.77-83]:

- спосіб вистави документів;
- спосіб вистави пошукових запитів;

- вид критерію релевантності документів.

Варіації цих складових визначають велика кількість усіляких реалізацій систем текстового пошуку.

Класичні моделі інформаційного пошуку розглядають документи як безлічі, що представляють ці документи ключових слів, надалі називаних термами. Терм - це звичайно просте слово, семантика якого допомагає описати основний зміст документа.

Формально, опис будь-якої моделі інформаційного пошуку складається з 4 частин [18]:

- ***D*** - безліч вистав документа;
- ***Q*** - безліч вистав інформаційної потреби (запиту);
- ***F***- засобу моделювання вистав документа, запитів і їхніх відносин;
- ***R(q,di)*** - функція ранжирування, яка парі документ/запит зіставляє деяке речовинне число.

Моделі інформаційного пошуку діляться на три класи:

• **Теоретико-множинні моделі** - у якості каркаса використовують теорію множин, коли документи й запити представляються у вигляді безлічі термів.

• **Імовірнісна модель** - каркасом для таких моделей виступає теорія ймовірностей. У якості оцінки релевантності документа запиту користувача використовується ймовірність того, що користувач визнає документ істинно релевантним.

• **Алгебраїчна модель** - документи й запити описуються у вигляді векторів у багатомірному просторі. Каркасом для таких моделей виступають алгебраїчні методи.

У рамках кожного із класів була запропонована безліч альтернативних моделей. Незважаючи на ряд недоліків, на практиці класичні теоретико-множинні моделі досить популярні в силу своєї простоти. Хоча імовірнісні моделі пропонують найбільш природній спосіб формально описати проблему інформаційного пошуку, їх популярність відносно невелика. Найбільш популярними є алгебраїчні моделі, оскільки їх практична ефективність звичайно

виявляється вище. Загалом кажучи, пропоновані останнім часом нові моделі інформаційного пошуку найчастіше є гібридними й мають властивості моделей різних класів.

2.8.1 Теоретико-множинні моделі

У рамках цього класу розглянемо булеву модель, що є класичним прикладом.

У даних моделях пошуку користувач може формулювати запит у вигляді булевого вираження, використовуючи для цього оператори И, АБО, НЕМАЄ. Терми запиту залежать від конкретного варіанта моделі пошуку:

- По тексту, термами будуть слова, критерієм релевантності буде умова входження деякого слова або словосполучення в текст документа.
- По класифікаторах, термами вираження будуть ідентифікатори класів класифікатора.
- З використанням дублінського ядра, термом буде значення елементів метаданих. Документ, що має співпадаючі значення елементів метаданих зі значеннями, заданими в запиті, уважається релевантним.

2.8.2 Імовірнісна модель

В 1977 році Robertson і Sparck-jones (Робертсон і Спарк- Джоунз) обґрунтували й реалізували імовірнісну модель (запропоновану ще в 1960 [5]).

Релевантність у цій моделі розглядається як імовірність того, що даний документ може виявитися цікавим користувачеві. При цьому мається на увазі наявність уже існуючого первісного набору релевантних документів, обраних користувачем або отриманих автоматично при якому-небудь спрощенім припущенні. Імовірність виявитися релевантним для кожного наступного документа розраховується на підставі співвідношення зустрічальності термінів у релевантному наборі й в інший, "нерелевантної" частини колекції.

Імовірність того, що документ d - релевантний P(R | d), визначається на основі теореми Байеса:

$$P(R \mid d) = \frac{P(d \mid R) \cdot P(R)}{P(d)} \quad (2.3)$$

- R - релевантність;
- P(R) - імовірність того, що випадково обраний з колекції документ D є релевантним;
- P(d | R) - імовірність випадкового вибору документа d з безлічі релевантних документів;
- P(d) - імовірність випадкового вибору документа d з колекції D;

Хоча імовірнісні моделі мають деяку теоретичну перевагу, адже вони розташовують документ у порядку убування "імовірності виявитися релевантним", на практиці вони так і не одержали великого поширення. Важливо помітити, що в кожному із сімейств найпростіша модель виходить із припущення про взаємонезалежність слів і має просту умову фільтрації: документи, що не містять слова запиту, ніколи не бувають знайденими. Просунуті ("альтернативні") моделі кожного із сімейств не вважають слова запиту взаємонезалежними, а, крім того, дозволяють знаходити документи, що не містять жодного слова із запиту.

2.8.3 Алгебраїчна модель

З даного класу розглянемо дві моделі векторну, що є класичним представником і латентно-семантичний аналіз.

Векторна модель була з успіхом реалізована в 1968 році батьком - засновником науки про інформаційний пошук Джерардом Солтоном (Gerard Salton) у пошуковій системі SMART (Salton's Magical Automatic Retriever of Text).

У цей час векторні моделі є найпоширенішими й застосовуваними на практиці моделями пошуку. Векторні моделі, на відміну від булевих, без праці дозволяють ранжувати результуюча безліч документів запиту.

Суть таких моделей зводиться до вистави документів і запитів у вигляді векторів.

Кожному терму ti у документі dj і запиті q зіставляється деяка ненегативна вага wij (wi для запиту). Таким чином, кожний документ і запит може бути представлений у вигляді до - мірного вектора:

$$dj = (w_{1j} w_{2j} w_{kj}), \qquad (2.4)$$

де до- к- загальна кількість різних термів у всіх документах.

Згідно з векторною моделлю, близькість документа di до запиту q оцінюється як кореляція між векторами їх описів. Ця кореляція може бути обчислена, наприклад, як скалярний добуток відповідних векторів описів [7, с.90-94].

Існують різні підходи до вибору зазначених ваг. Одним з найпростіших є використання нормалізованої частоти даного терма в документі:

$$w_{ij} = \frac{n_{ij}}{N_j}, \qquad (2.5)$$

де nij - кількість повторень даного терма в документі;

Nj- - загальна кількість усіх термів у документі.

Більш складні варіанти розрахунків ваг ураховують частоту використання даного терма в інших документах колекції, тобто враховують дискримінаційну силу терма. Але ці варіанти можливі тільки при наявності статистики використання термів у колекції.

Варіації всіляких способів призначення ваг термів і оцінки заходу близькості векторів визначають широкий спектр різних модифікацій даної моделі пошуку.

Здатність знаходити й ранжувати документи, що не містять слів із запиту, часто вважають ознакою штучного інтелекту або пошуку за змістом і відносять апріорі до переваг моделі.

Латентно-семантичний аналіз (LSA) - це теорія й метод для витягу контекстно-залежних значень слів за допомогою статистичної обробки більших наборів текстових даних. ПРОТЯГОМ декількох останніх років цей метод не раз використовувався як в області пошуку інформації, так і в завданнях фільтрації й класифікації [11].

Латентно-семантичний аналіз ґрунтується на ідеї, що сукупність усіх контекстів, у яких зустрічається й не зустрічається дане слово, задає безліч обопільних обмежень, які в значній мірі дозволяють визначити подібність значеннєвих значень слів і безліччю слів між собою.

У якості вихідної інформації LSA використовує матрицю терми-на-документи, що описує використовуваний для навчання системи набір даних. Елементи цієї матриці містять частоти використання кожного терма в кожному документі. Найпоширеніший варіант LSA заснований на використанні розкладання матриці за сингулярними значенням (SVD). Використовуючи SVD, величезна вихідна матриця розкладає в безліч із k, звичайно від 70 до 200, ортогональних матриць, лінійна комбінація яких є непоганим наближенням вихідної матриці. Більш формально, згідно з теоремою про сингулярне розкладання, будь-яка речовинна прямокутна матриця X може бути розкладена в добуток трьох матриць:

$$X = U\Sigma V^T \tag{2.6}$$

таких, що матриці U і V - ортогональні, а - діагональна матриця, значення на діагоналі якої називаються сингулярними значеннями матриці X.

Таке розкладання має чудову особливість: якщо в залишити тільки k найбільших сингулярних значень, а в матрицях U і V тільки відповідні до цих значень стовпці, то добуток матриць, що вийшли Σ_{isa}, U_{isa} і V_{isa} буде найкращим наближенням вихідної матриці X матрицею рангу *к*:

$$\Sigma \cong \bar{X} = U_{isa}\Sigma_{isa}V_{isa} \tag{2.7}$$

Основна ідея латентно-семантичного аналізу полягає в тому, що якщо в якості X використовувалася матриця терми-на-документи, то матриця утримуюча тільки до перших лінійно незалежних компонентів X, відбиває основну структуру асоціативних залежностей, що присутні у вихідній матриці, і в той же час не містить шуму.

Таким чином, кожний терм і документ представляються за допомогою векторів у загальному просторі розмірності до (так званім просторі гіпотез). Близькість між будь-якою комбінацією термів і/або документів може бути легко обчислена за допомогою скалярного добутку векторів.

Вибір найкращої розмірності до для LSA - відкрита дослідницька проблема. В ідеалі, до повинне бути досить велике для відображення всієї реально існуючої структури даних, але в той же час досить мало, щоб не захопити випадкові й маловажні залежності. Якщо обране до занадто велике, то метод втрачає свою міць і наближається по характеристиках до стандартних векторних методів. Занадто мале до не дозволяє вловлювати відмінності між схожими словами або документами. Дослідження показують, що з ростом до якість спочатку зростає, а потім починає падати [11].

Поява Internet не тільки стимулювало ріст інтересу до теорії інформаційного пошуку, але також обумовила появу нових і зміну постановки вже існуючих завдань пошуку.

Висновки до 2 розділу

Проведений аналіз сучасних пошукових систем показав велику кількість інновацій у цьому напрямку але усе ж таки завдання не виконується на 100%. Можливість оптимізації потребує застосування нових ефективніших методів.

1 Використання для більш релевантної видачі результатів комплексу методів фільтрування отриманої інформації.

2 Проведений аналіз ПС показав, що в них застосовують алгоритми індексування та ранжирування сторінок (Pagerank (Google)), інноваційні механізми пошуку та інформаційно-пошукові математичні моделі.

РОЗДІЛ 3
АНАЛІЗ МЕТОДУ ОПТИМІЗАЦІЇ

3.1 Використання лінгвістичного процесор для інформаційно-пошукової системи

3.1.1 Лінгвістичний процесор ЕТАП

Спроби формалізувати інтелектуальну діяльність людини привели до постановки фундаментального лінгвістичного завдання, що полягає в моделюванні його язикової поведінки, тобто в побудові функціональної кібернетичної моделі природної мови. Природна мова служить людині для вираження власних думок і для розуміння думок інших людей. З відомим огрубінням можна сказати, що першому виду язикової діяльності відповідає виробництво текстів на ПМ, а другому - розуміння таких текстів. Якщо позначити безліч текстів через {Т} а безліч змістів, що виражаються ними, через {З}, то модель ПМ можна визначити як транслятор, що встановлює відповідність (1) між цими двома множинами:

$$\{Т\}\leftrightarrow\{З\} \qquad (3.1)$$

Формальні моделі мови, що розроблялися спочатку в чисто теоретичному плані, останнім часом усе частіше розглядаються як компоненти різних прикладних систем. Будучи реалізовані на комп'ютері, вони входять як складені частини в системи машинного перекладу (МП), підсистеми спілкування з базами даних (БД) на необмеженому ПМ й інші складні інформаційні системи.

Будемо називати комп'ютерну систему, що реалізує формальну лінгвістичну модель і здатну працювати з ПМ у всьому його обсязі, лінгвістичним процесором (ЛП). У сучасній інформатиці лінгвістичними процесорами називаються й інші засоби переробки текстової інформації на ПМ, у тому числі й не розраховані на роботу з ПМ в повному обсязі.

Дві основні функції ЛП полягають у витягу змісту із заданого тексту (моделювання розуміння, аналіз) і у вираженні заданого змісту текстом на ПМ (моделювання виробництва текстів, синтез).

Звичайно говорять про розуміння текстів у слабкому й сильному змісті. Розуміння в слабкому змісті має місце тоді, коли оброблюваний текст може бути перифразований засобами того ж самого або іншого мови. Ця модель розуміння реалізується, наприклад, при перекладі тексту з один ПМ на іншій: переклад є переказ змісту тексту, написаного на одній мові, засобами іншої мови. Розуміння в сильному змісті має місце тоді, коли сприйманий текст вимагає від адресата певної реакції, і коли ця реакція дійсно має місце. Ця модель розуміння реалізується, наприклад, при спілкуванні із БД, коли остання видає правильна відповідь на звернений до неї питання.

Для моделювання розуміння в обох зазначених змістах у будь-яких системах природно-язикових текстів, що претендують на глибоку переробку, необхідно . мати особливий рівень вистави висловлень, який можна назвати семантичним. Він задається формальною семантичною мовою, виразні засоби якого досить великі для того, щоб відбити повністю зміст тексту на вихідному ПМ в рамках поставленого завдання.

Розташовуючи такою мовою, можна формально описати процес розуміння (аналізу) текстів як переклад із природної мови на семантичну, а процес виробництва текстів (синтез) - як "зворотний" переклад із семантичної мови на природну. Про те, як розуміється формальна семантична мова, ми скажемо нижче. Що стосується тексту на ПМ, то під ним розуміється послідовність пропозицій у звичайному орфографічному записі. Не передбачається здійснювати розпізнавання й породження звучної мови, хоча в принципі ЛП може бути доповнений відповідними фонетичними компонентами для того, щоб на вхід аналізу надходив вимовний мовний фрагмент, а синтез закінчувався озвученим читанням отриманого тексту на ПМ.

Принципова новизна створюваного ЛП полягає в тому, що формальна модель мови, що лежить у його основі, є найбільш повною моделлю такого роду. Це модель класу "Зміст → Текст". Така модель забезпечує одержання зв'язних синтаксичних структур для всіх пропозицій оброблюваних текстів,

незалежно від ступеня їх складності, і переробку текстів природною мовою без значеннєвих втрат.

Робочі мови у відповідних проектах - англійський і німецький. Ці проекти уступають справжньому проекту в силу того, що не опираються на принципову лінгвістичну модель і тим самим споконвічно не в змозі забезпечити повноту й надійність обробки природно-язикових даних. Крім того, вони не працюють із російською мовою. Таким чином, даний Ж є пріоритетною розробкою в обох зазначених змістах.

3.1.2 Основні принципи розробки

При розробці ЛП для складних інформаційних систем ми керувалися наступними засадами [16].

1. Незалежність форматів завдання лінгвістичних знань цієї мови, яка забезпечує можливість їх наповнення даними будь-яких мов і перетворення будь-яких систем у багатомовні.

2. Незалежність граматик і словників від алгоритмів, яка забезпечує "відкритість" лінгвістичних знань і можливість зручного коректування граматик і словників. По суті цей принцип означає "декларативність" завдання лінгвістичної інформації. Він послідовно проведений у всій системі, за винятком частини алгоритму синтаксичного аналізу пропозиції: поряд з повним алгоритмом синтаксичного аналізу для прискорення роботи використовується алгоритм фрагментного аналізу, у рамках якого істотна частина необхідних для нього лінгвістичних знань задається процедурно.

3. Незалежність лінгвістичних знань від предметної області, завдяки якій забезпечується можливість адаптації єдиних алгоритмів аналізу й синтезу текстів для обробки даних з нових предметних областей.

4. Незалежність лінгвістичних знань від характеру розв'язуваного завдання, яке дозволяє використовувати ту саму формальну модель мови для розв'язку всіляких завдань, наприклад, таких, як спілкування із БД на ПМ, з одного боку, і МП - з іншої. Лінгвістичний процесор, що задовольняє цій

вимозі, виявляється здатним обслуговувати багато типи складних інформаційних систем, тобто здобуває риси полі-функціональності.

Виконання двох останніх умов означає, що формальна модель мови, що лежить в основі ЛП, повинна бути теоретично обґрунтованої (побудованої з обліком усіх новітніх досягнень лінгвістичної науки) і повної. ЛП повинні бути доступні знання ПМ в обсязі, у якім їм володіють його носії. Практично це значить, що ЛП повинен бути здатний обробляти будь-які морфологічні явища даного ПМ, усі синтаксичні конструкції ділової прози й близько 10 000 слів загального поширення, які зустрічаються в будь-яких типах текстів даною мовою. При цьому ЛП повинен однаково добре володіти як зовнішньою стороною всіх перерахованих язикових засобів, так і їх семантикою. Тільки така принципова модель мови може забезпечити переробку довільних текстів, не вимагаючи корінної перебудови при перехід до кожного нового типу текстів.

Сучасний стан лінгвістичних знань дозволяє будувати стовідсотково повні й надійні моделі морфології. Достатньому ступеня повноти й надійності можна досягти й у моделі синтаксису. Зовсім реальною перспективою є створення словників обсягом до декількох десятків тисяч слів, що забезпечують високоякісний морфологічний і синтаксичний аналіз і синтез текстів ділової прози практично необмеженому ПМ. Це відкриває можливість зробити морфологічний і синтаксичний компоненти процесора універсальними, тобто не залежними від характеру конкретного прикладного завдання. Вони можуть використовуватися в широкому спектрі інформаційних систем, включаючи системи автоматичного розуміння текстів, МП, спілкування із БД і т.п., тобто дійсно здобувають властивості полі-функціональності. Якщо ж говорити про семантику, то у своєму нинішньому стані " вона ще не готова до побудови універсальних моделей свого об'єкта, тобто моделей, що враховують настільки широке коло явищ ПМ, що вони здатні обслуговувати будь-які прикладні завдання. Тому модель семантики в нашому ЛП була із самого початку орієнтована на конкретну проблему. Про те, як це було зроблено.

3.1.3 Використовувані компоненти

Основні елементи лінгвістичного й логіко-алгоритмічного забезпечення ЛП:

а) форма вистави МОРФС, СИНТС і СЕМС;

б) моделі морфологічного аналізу й синтезі (для російського й англійського мов);

в) моделі синтаксичного аналізу й синтезу (для російського й англійського мов), кожна з яких нараховує до 500 - 600 правил; для прискорення роботи алгоритмів синтаксичного аналізу й синтезу всі правила розділені на загальні, що включаються в обробку будь-якої пропозиції, до словникові, які активуються певними словами, якщо вони зустрілися в пропозиції; словникові правила діляться на трафаретні, застосовні до замкненого класу слів, і властиво словникові, застосовні тільки до даного слова й записувані в його словниковій статті;

г) модель семантичного аналізу (для російської мови), що нараховує близько 100 правил;

д) масиви правил перекладу з англійської мови на російську й з російської мови на англійську, що нараховують до 500 загальних і трафаретних правил кожний, і велике числі властиво словникових правил перекладу;

е) морфологічні словники російського й англійського мов; перший з них нараховує до 12 000 слів, а другий - до 15 000; другий словник використовується в підсистемі швидкого машинного перекладу, яка видає послівний переклад оброблюваного пропозиції на російську мову разом з його морфологічною структурою;

ж) англійський і росіянин комбінаторні словники, призначені для систем англо-росіянина й російсько-англійського автоматичного перекладу, обсягом близько 10 000 одиниць кожний;

з) семантичний словник для російської мови, призначений для перекладу запитів на мову SQL.

В області логіко-алгоритмічного забезпечення:

а) формальні мови для запису лінгвістичної інформації; усі лінгвістичні знання, використовувані в ЛП, записані на цих формальних мовах;

б) алгоритми морфологічного аналізу й синтезу, що забезпечують перетворення пропозицій на ПМ в них МОРФС і назад;

в) алгоритми передсинтаксичного й синтаксичного аналізу текстів, що одержують на вході МОРФС оброблюваного пропозиції, що й видають на виході його СИНТС;

г) алгоритм перетворення деревних структур, що забезпечує послідовне застосування до структури всіх необхідних лінгвістичних правил її перетворення для одержання СИНТС вихідної мови (у системах МП) або СЕМС (у системі спілкування із БД на ПМ);

д) алгоритм перекладу СЕМС у формулу мови SQL.

Забезпечення трансляції лінгвістичних даних у машинну форму й реалізація всіх згаданих вище алгоритмів.

3.1.4 Структура і склад

З боку свого внутрішнього обладнання ЛП являє собою багаторівневий перетворювач. У ньому різняться три рівні по-фразного вистави тексту - морфологічний, синтаксичний і семантичний. Кожний з рівнів обслуговується відповідним компонентом моделі - масивом правил і певним словником або словниками. На кожному з рівнів пропозиція має формальний образ, іменований надалі його структурою - морфологічної (МОРФС), синтаксичної (СИНТС) і семантичної (СЕМС). У цілому роботу моделі при аналізі можна представити наступною блок-схемою:

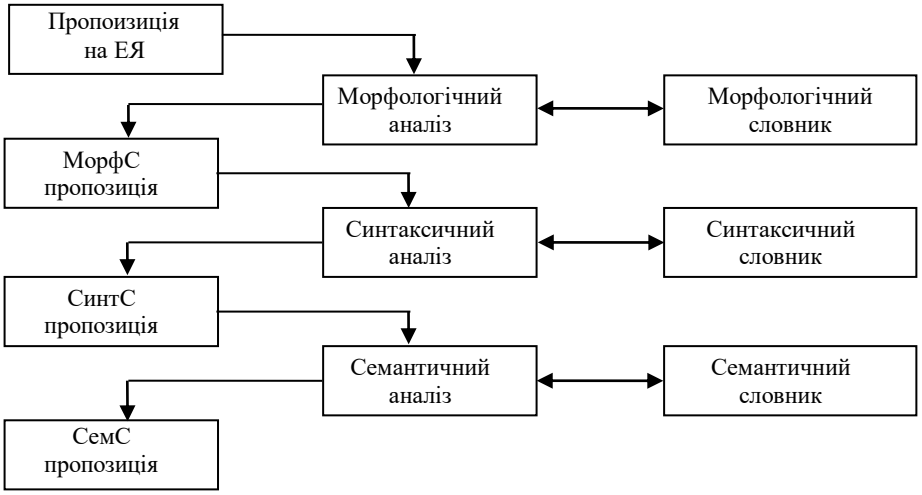

Рисунок 3.1 – Робота моделі при аналізі

Синтез являє собою зворотній перехід від СЕМС пропозиції до його запису у звичайному орфографічному виді. Оскільки процедура аналізу, у всякому разі "з погляду " комп'ютера, суттєво складніше процедури синтезу й свідомо містить у собі всі засоби, необхідні для цієї останньої, ми надалі зосередимося переважно на аналізі [11].

Під морфологічною структурою розуміється послідовність вхідних в аналізовану пропозицію слів із вказівкою частини мови й морфологічних характеристик (падежу, числа, роду, натхненності, часу, виду й т.п.).

Під синтаксичною структурою розуміється дерево залежностей, у вузлах якого коштують слова даної природньої мови із вказівкою частини мови й граматичних характеристик, а дуги відповідають специфічним для даної природньої мови відносинам синтаксичного підпорядкування. В описуваній моделі використовується 40-60 різних відносин; російський синтаксис описується за допомогою 55 відносин [20].

Під семантичною структурою розуміється дерево залежностей, у вузлах якого коштують або предметні імена, або слова універсальної семантичної мови (наприклад, імена таблиць, у яких зосереджені відомості про дану предметну область, атрибути таблиць, операторні символи), а дуги відповідають універсальним відносинам семантичного підпорядкування, таким, як

аргументне, атрибутивне, кон`юнкція, диз'юнкція, рівність, нерівність, більше, менше, належить, не належить і т.п. Істотним компонентом СЕМС є інформація про кореферентності вузлів, тобто інформація про те, у яких випадках мова йде про один і той самий об'єкт, а в яких - про різних.

Лінгвістичний процесор у цілому повинен забезпечувати виконання наступних перетворень: пропозиція на ПМ → МОРФС → СИНТС → СЕМС (при аналізі), СЕМС → СИНТС → МОРФС → пропозиція на ПМ (при синтезі).

Вище ми вже говорили, що в даній монографії основна увага буде приділена аналізу як більш важкої із цих двох процедур. Додамо до цього, що саме процедура аналізу забезпечена всіма необхідними для її виконання типами правил. Що стосується процедури синтезу, те лінгвістично етап СЕМС → СИНТС ще не забезпечений.

Отже, щоб побудувати ЛП зазначеного типу, необхідно розробити 1) формальні мови для запису (образів) пропозицій на морфологічному, синтаксичному й семантичному рівнях вистави; 2) формальне поняття структури пропозиції для кожного із цих рівнів; 3) масиви правил для перетворення структур суміжних рівнів друг у друга; 4) морфологічний, комбінаторний і семантичний словники, включивши в них усю інформацію про кожну лексему, необхідну для здійснення відповідного перетворення [21].

Щоб одержати багатомовний Ж, таку роботу слід виконати для кожного із що брав участь у ньому ПМ.

Нарешті, щоб одержати полі-функціональний ЛП, необхідно постійно поповнювати його засобами розв'язку кожного чергового завдання, якщо вони специфічні для неї. Так, для автоматизації перекладу з один ПМ на інший ЛП повинен бути доповнений відповідним масивом правил перекладу [23].

3.1.5 Опис алгоритму

У блоці морфологічного аналізу присутні не тільки функції морфологічного аналізу, але й засобу підтримки й редагування словникових файлів, необхідних аналізатору.

Працює морфологічний аналізатор у такий спосіб. На його вхід надходить масив "слів", розділових знаків і чисел, виділених із вхідного тексту на етапі лексичного аналізу. Для кожного "слова" аналізатор виконує процедуру пошуку в словнику основ, завантаженому на згадку. При цьому шукаються всі основи, з яких може починатися аналізоване слово. Якщо чергова основа задовольняє цій умові, то зі словника афіксів витягується рядок, що містить усі можливі афікси для даної основи. Кожний афікс із цього рядка по черзі приєднується до основи, і результат рівняється з аналізованим словом. У випадку їх точного збігу формується черговий запис у список результатів пошуку: по порядковому номеру афікса в рядку афіксів визначаються змінні морфологічні параметри слова (наприклад, для іменника - число й падіж), а за словниковою інформацією даної основи - його постійні параметри (для іменника - рід і натхненність) [17].

Якщо в результаті такого пошуку не знайдене жодного успішного варіанта, то проводиться пошук серед виключень. Виключення присутні в словнику основ поряд зі звичайними основами. І ті, і інші мають у словнику інформацію про постійні морфологічні ознаки й про номер рядка припустимих афіксів.

Різниця між виключеннями й звичайними основами полягає в тому, що, по-перше, рядок з незмінною частиною слова у виключень порожня, і, по-друге, номер рядка афіксів для виключень ставиться не до файлу афіксів, а до окремого файлу виключень. Структура цього файлу точно така ж, але в нього внесені цілі словоформи, а не їхні закінчення. Таким чином, при пошуку серед виключень доводиться переглядати всі словоформи всіх присутніх у словнику виключень. Це займає багато часу, тому пошук серед виключень проводиться тільки в тому випадку, коли не знайдене жодного варіанта серед звичайних основ. Сам аналіз проводиться точно так само. Якщо деяка словоформа деякого виключення точно збігається з аналізованим словом, то по номеру словоформи визначаються змінні морфологічні параметри слова, а за словниковою інформацією самого виключення - постійні параметри слова.

Якщо після пошуку серед виключень однаково не знайдене жодного варіанта, то перевіряється наявність в аналізованого слова поворотного суфікса "-СЯ", "-СЬ", або приставок "НЕ-", "НІ-". Якщо вони є, то вони відтинаються від аналізованого слова, і процедура пошуку повторюється спочатку. При цьому морфологічні параметри основ, що перебувають, модифікуються спеціальною процедурою [21].

У випадку, коли всі етапи пошуку дали негативний результат (не знайдене жодного варіанта), користувачеві видається запит на введення нової основи в словник. У випадку його відмови це зробити виконання морфологічного аналізу припиняється. Якщо ж нове слово введене в словник, то вся процедура пошуку повторюється спочатку.

Для чисел і розділових знаків аналізатор формує такі ж запису, як і для слів, але в якості морфологічних ознак затягає в них або порядковий номер розділового знака, або тип числівника, у який може бути перетворене вихідне число.

3.1.6 Структура програмного застосуванні

Лінгвістичний процесор виконує роль посередника між користувачем і базою даних, у якій зберігається його інформація, що цікавить. Завданням ЛП є перетворення природно-язикової пропозиції в деякий набір семантичних структур, що є формальною виставою "змісту" вихідної пропозиції або тексту. ЛП виконаний у вигляді бібліотеки, доступної різним додаткам і не взаємодіючої прямо з користувачем. При цьому загальна структура програмного продукту представляється так:

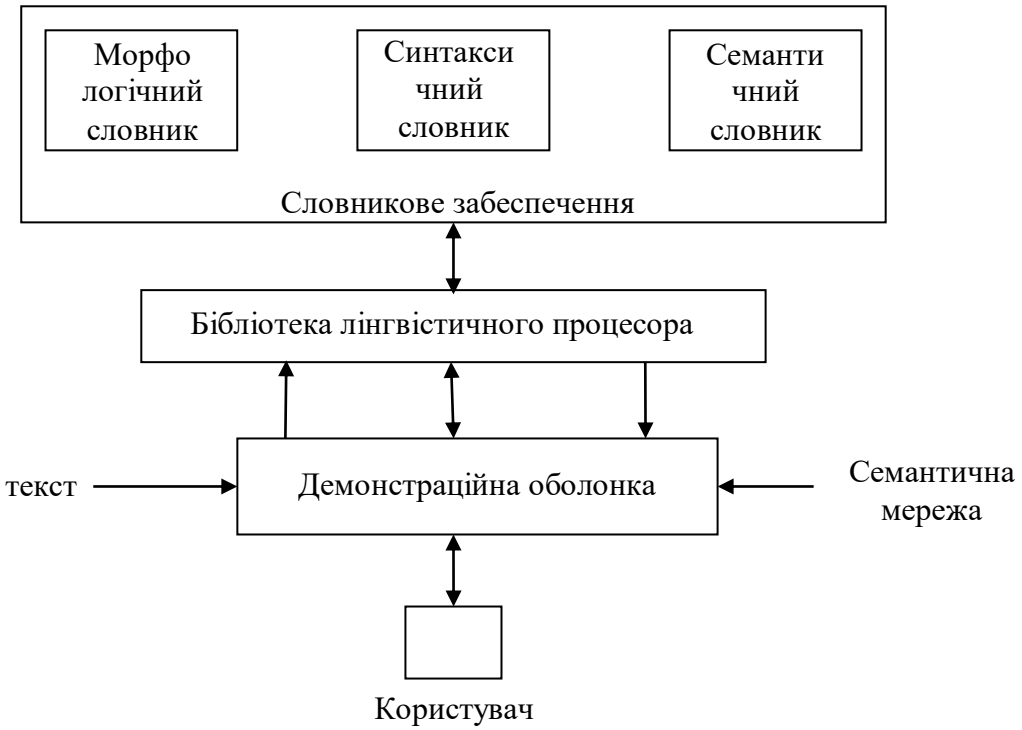

Рисунок 3.2 – Структура програмного застосування

Блок лексичного аналізу

У даному ЛП використовується найпростіший лексичний аналізатор; він виконує допоміжні функції, не має можливості настроювання або взаємодії з користувачем і тому реалізований як частина блоку морфологічного аналізу.

Блок лексичного аналізу ухвалює вихідний текст безпосередньо від елементів користувацького інтерфейсу - а саме, від текстового редактора. Аналізована пропозиція попадає на вхід лексичного аналізатора у вигляді масиву Ascii-Символів, що містить прописні й малі літери російського алфавіту, цифри, знаки пунктуації [19]:

Разом з покажчиком на масив символів у процедуру лексичного розбору передається змінна - лічильник байт у цьому масиві. Отриманий масив аналізатор повинен перетворити в масив лексичних одиниць. (Тут під цим терміном мається на увазі слово, число або розділовий знак.) Для кожної лексичної одиниці формується окремий рядок, у який копіюються всі символи,

що належать даній лексичній одиниці. При цьому віддаляються пробіли, символи переносу, кінця рядка й незнайомі символи:

Покажчики на всі сформовані в такий спосіб рядка аналізатор затягає у вихідний динамічний масив, який є результатом його роботи.

Блок морфологічного аналізу

Серед методів морфологічного аналізу, що використовуються в лінгвістичних процесорах, можна виділити методи з декларативної й із процедурною орієнтацією. Для методів декларативної орієнтації характерна наявність повного словника всіх можливих словоформ для кожного слова. При цьому кожна словоформа забезпечується повною й однозначною морфологічною інформацією, куди входять як постійні, так і змінні морфологічні параметри. Завдання морфологічного аналізу в цьому випадку зводиться до пошуку потрібної словоформи в словнику й копіюванню морфологічної інформації, відповідній до знайденої словоформи, у програму [40].

У процедурних методах кожне слово розділяється на основу й афікс (закінчення й, можливо, суфікс), і словник містить тільки основи слів разом з посиланнями на відповідні рядки в таблиці можливих афіксів. Основний критерій при розбивці слова на основу й афікс - основа повинна залишатися незмінної у всіх можливих словоформах даного слова. Оскільки велика кількість слів російської мови має ті самі афікси, те сумарний обсяг словника основ і словника афіксів виявляється значно менше, чим обсяг повного словника всіх словоформ, використовуваного в декларативних методах. Однак процедура морфологічного аналізу ускладнюється: тепер зі словника основ необхідно по черзі вибирати всі основи, що збігаються з початковими буквами аналізованого слова, і для кожної такої основи перебирати всі можливі для неї афікси. У випадку точного збігу чергового варіанта "основа+афікс" з аналізованим словом варіант аналізу вважається успішним, і в програму передається морфологічна інформація, відповідна до даної основи й даному

афіксу. При цьому, як правило, постійні морфологічні параметри визначаються основою слова, а змінні – афіксом [39].

Іноді використовується комбінований варіант морфологічного аналізу. При цьому використовується як словник словоформ, так і словник основ. На першому етапі проводиться пошук по словникові словоформ, і у випадку успішного пошуку аналіз на цьому завершується. А якщо ні, то задіється словник основ і процедурний метод аналізу.

Основним недоліком декларативних методів є надмірно великий обсяг словника, що породжує ряд технічних проблем [41]:

- більші витрати праці на створення й підтримка словника;
- неможливість повного розміщення словника в оперативній пам'яті комп'ютера при аналізі;
- висока надмірність інформації, пов'язаної з постійними ознаками кожної словоформи (морфологічними, синтаксичними, семантичними);

Перевагами методу є простота (і, як наслідок, висока швидкість) аналізу, а також універсальність стосовно безлічі всіх можливих словоформ російської мови.

Для процедурних методів час аналізу одного слова може бути суттєво вище, але обсяг використовуваних словників у невеликих системах дозволяє завантажувати словники цілком в оперативну пам'ять. Крім того, такі словники значно легше створювати, оскільки постійні параметри кожного слова вводяться однократно, разом з його основою. Істотним недоліком процедурних методів є відсутність універсальності. Інакше кажучи, існує велика кількість слів, які не можна представити у вигляді суми незмінної основи й афікса. (Наприклад, іменник "рік", яке має в множині родового відмінка форму "років"; займенник "я" і т.д.). У розглянутому ЛП застосований процедурний метод морфологічного аналізу, доповнений механізмом обробки таких виключень.

Предикати дії

З безлічі слів російської мови по семантичних ознаках можна виділити наступні категорії:

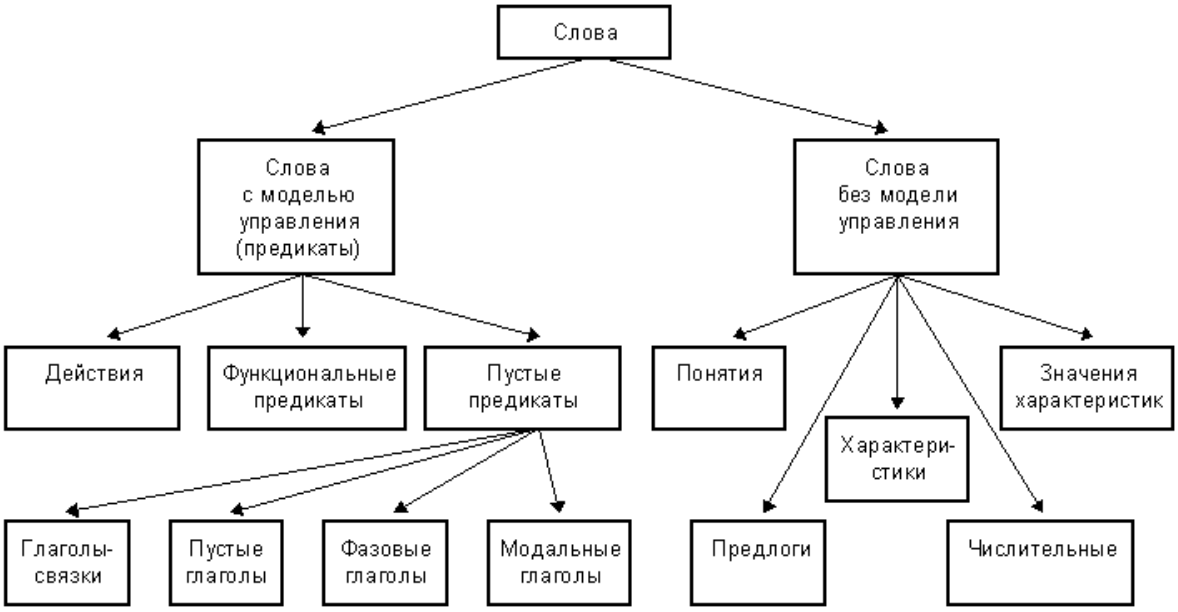

Рисунок 3.3. – Класифікація слів російської мови по семантичних категоріях

Для лінгвістичного процесора, призначеного для роботи в складі ІПС, найбільш важливої є обробка предикатів дії і їх можливих актантів - понять, характеристик і їх значень, а також числівників. Обробка функціональних і порожніх предикатів може бути зведена до модифікації фрагментів семантичної мережі, побудованих для залежних від них предикатів дії.

Кожний предикат має одну або кілька моделей керування (МК). Модель керування накладає синтаксичні й семантичні умови на можливі актанти (аргументи) даного предиката й указує їхні семантичні ролі стосовно предиката. У спрощеному виді МК можна представити як таблицю, кожний рядок у якій визначає один з можливих актантів. Цей рядок містить: привід (якщо є), частина мови й падіж актанта, його семантичне мета-поняття (категорію), семантичну роль у предикаті й ознака обов'язковості даної ролі.

Аналіз моделей керування дозволяє природно перейти від синтаксичного дерева залежностей до фрагмента семантичної мережі. Для цього в найпростішому випадку досить: 1) для кожного предиката відшукати в дереві залежностей усі присутні актанти; 2) створити по одній вершині на кожний актант і ще одну - для самого предиката; 3) провести від вершини-предиката

дугу до кожної вершини-актанту; при цьому ім'я дуги (її семантична роль) вибирається з моделі керування залежно від актанта. Якщо модель керування не допускає такого актанта в жодній семантичній ролі, залишається дві можливості: або дана ТЕ у дійсності є стосовно предиката не актантом, а обставиною (тоді цей зв'язок можна відобразити дугою відповідного типу), або має місце помилка синтаксичної структури [33].

Правильно розрізняти ці два випадки вдається не завжди, однак можна вказати умова, при якім з великою ймовірністю присутня саме помилка синтаксичної структури. Мається на увазі ситуація, коли актант задовольняє морфологічним умовам моделі керування, а семантичним - немає. Оскільки синтаксичний аналізатор при угрупованні не перевіряє семантичні ознаки, він не може самостійно виявляти помилки такого типу. Таким чином, якщо помилка виявлена на етапі семантичного аналізу, необхідно повернутися назад до синтаксичного аналізу й спробувати знайти інший варіант синтаксичної структури, який би не містив невірного актанта в піддереві залежностей даного предиката. Для того щоб синтаксичний аналізатор не намагався знову об'єднати предикат з невірним актантом, у Те-Предикаті й Те-Актанте встановлюється спеціальний прапор помилки.

У підсумку обробка предикатів дії може бути представлена у вигляді наступної схеми:

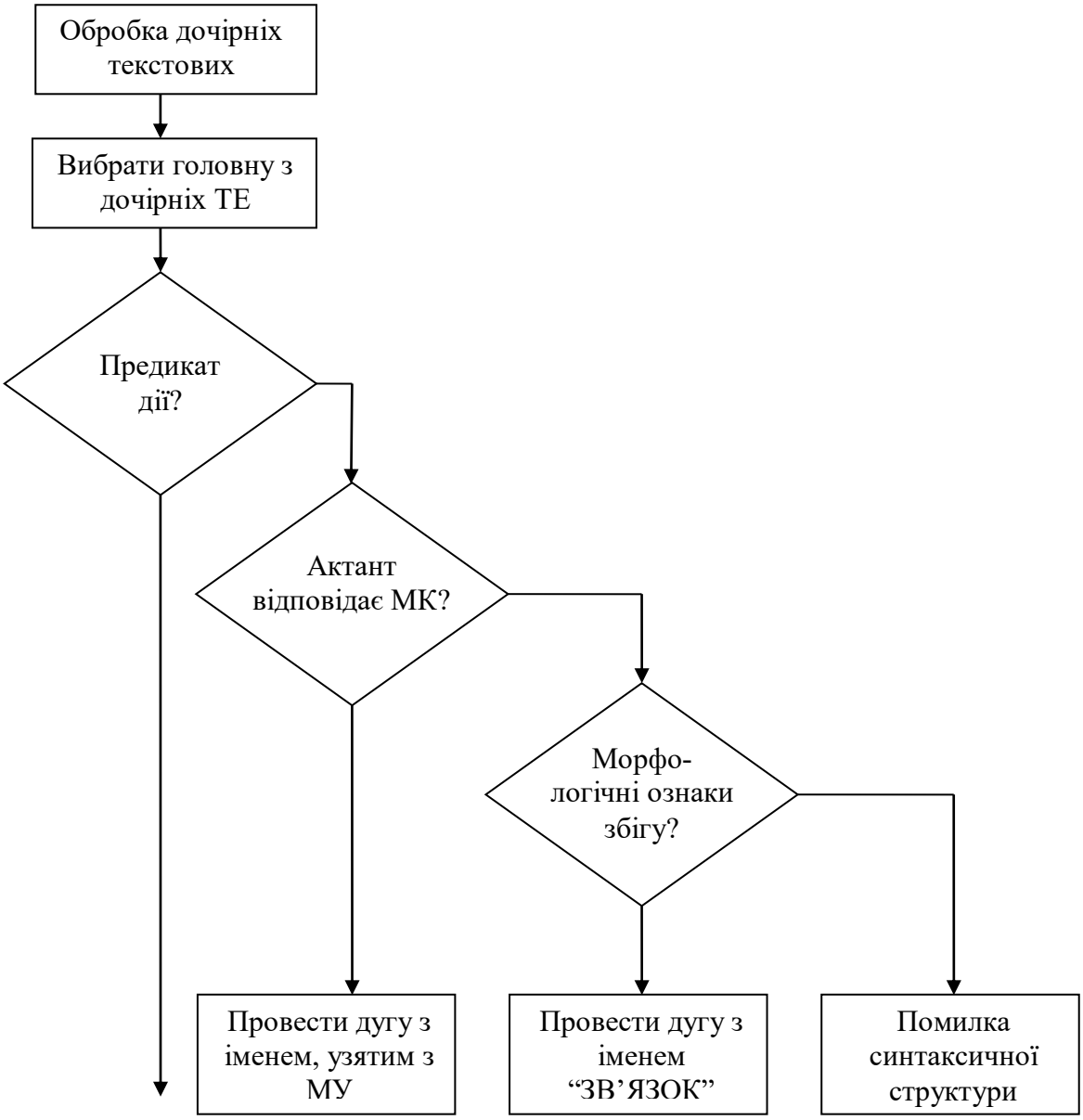

Рисунок 3.4 – Блок-схема обробки предикатів дії

3.2 Прикладні функції

Описуваний ЛП може бути безпосередньо використаний принаймні у двох типах прикладних систем [37]:

1) у системі спілкування з комп'ютером на практично необмеженому ПМ. Подібна система дозволяє організувати максимально дружній інтерфейс користувача з комп'ютером, оскільки не вимагає або майже не вимагає від користувача попередніх знань в області техніки й експлуатації ЕОМ;

2) у системах МП науково-технічних текстів і ділової документації з іноземних мов на російську й з російської на іноземні.

У принципі можливе перетворення системи спілкування на ПМ в багатомовну шляхом її сполучення із системою МП.

Обидві названі завдання ставляться як завдання перекладу, з тою різницею, що в першому випадку мова йде про переклад з ПМ на штучну мову, а саме, на формальну мову запитів, прийнятий у даній системі керування базами даних (СУБД), а в другому - про переклад з один ПМ на іншій.

Є ще два класи завдань, розв'язних у більш віддаленій перспективі, для яких ЛП може виявитися корисним у повному своєму обсязі.

Перший із цих класів завдань пов'язаний з автоматичним поповненням баз даних безпосередньо по текстах. Для розв'язку таких завдань необхідні, крім властиво лінгвістичного процесора, здатного розуміти тексти даної предметної області, ще й логічні процесори. Вони повинні бути оснащені таким набором функцій, за допомогою яких можна порівнювати вже наявну в БД інформацію зі знову вступники й витягати з оброблюваного тексту принципово нову інформацію,, якщо вона в ньому втримується.

Другий клас завдань ставиться до "планування тексту". Якщо в завданнях автоматичного поповнення БД по текстах ЛП використовується як аналізатор, то в завданнях планування тексту на перший план виходять його активні - синтезуючі або текстопороджуючі - функції. Одна з типових конкретних завдань цього роду - породження опису на ПМ поточної роботи СУБД по таких параметрах, як дати й часи її використання, імена користувачів, яким були зроблені інформаційні послуги, істота цих послуг (що саме цікавило користувача), імовірні цілі спрощення до СУБД і т.п. У цім завданні теж необхідне сполучення ЛП із логічними процесорами. При цьому спочатку працюють саме логічні процесори, що породжують план тексту, а потім - його розгорнуту концептуальну структуру. Після цього вступає в дію ЛП, який спочатку перетворить концептуальну структуру в семантичну структуру майбутнього тексту на ПМ, а потім, через ряд проміжних етапів, перетворює її в послідовність пропозицій, що утворюють зв'язну розповідь на задану користувачем тему [26].

В обох зазначених класах завдань передбачається використання ЛП у повному обсязі. Цим його можливі функції й застосування не вичерпуються. Різні елементи й модулі лінгвістичного процесора, насамперед лінгвістичні знання, що втримуються в ньому, допускають використання й в інших інформаційних системах, насамперед, у партнерських системах типу "помічник вчителя" (російської або іноземної мов), а також у різного роду комп'ютерних словниках.

Розглянемо більш докладно два завдання, сформульовані на початку сьогодення роздягнула, для принципового розв'язку яких досить засобів самого ЛП у його нинішньому виді, завдання спілкування з базами даних природною мовою й завдання машинного перекладу.

Залежно від характеру конкретного розв'язуваного завдання переробка пропозиції доводить до більшого або меншого рівня глибини. Так, у завданні спілкування із БД на необмеженому ПМ необхідно зважати на те, що формальний образ запиту мовою СУБД може разюче відрізнятися від вихідної пропозиції на ПМ.

3.2.1 Взаємодія блоків ЛП при аналізі пропозиції

Лінгвістичний процесор не має засобів взаємодії з користувачем; такі засоби повинне надавати програмне оточення ЛП. Також у завдання ЛП не входить пряма взаємодія з базами даних (БД вихідних документів і БД результатів аналізу) - цю функцію буде виконувати прикладна програма, що використовує ЛП.

На схемі показані основні програмні блоки ЛП (прямокутники), їхня словникова підтримка (пунктирні прямокутники), взаємодія з користувачем (стилізовані діалогові вікна), а також структури даних, передані між програмними блоками. Суцільними лініями показаний нормальний хід аналізу, пунктирними - дії при помилках.

Нижче представлена загальна схема роботи лінгвістичного процесора російської мови:

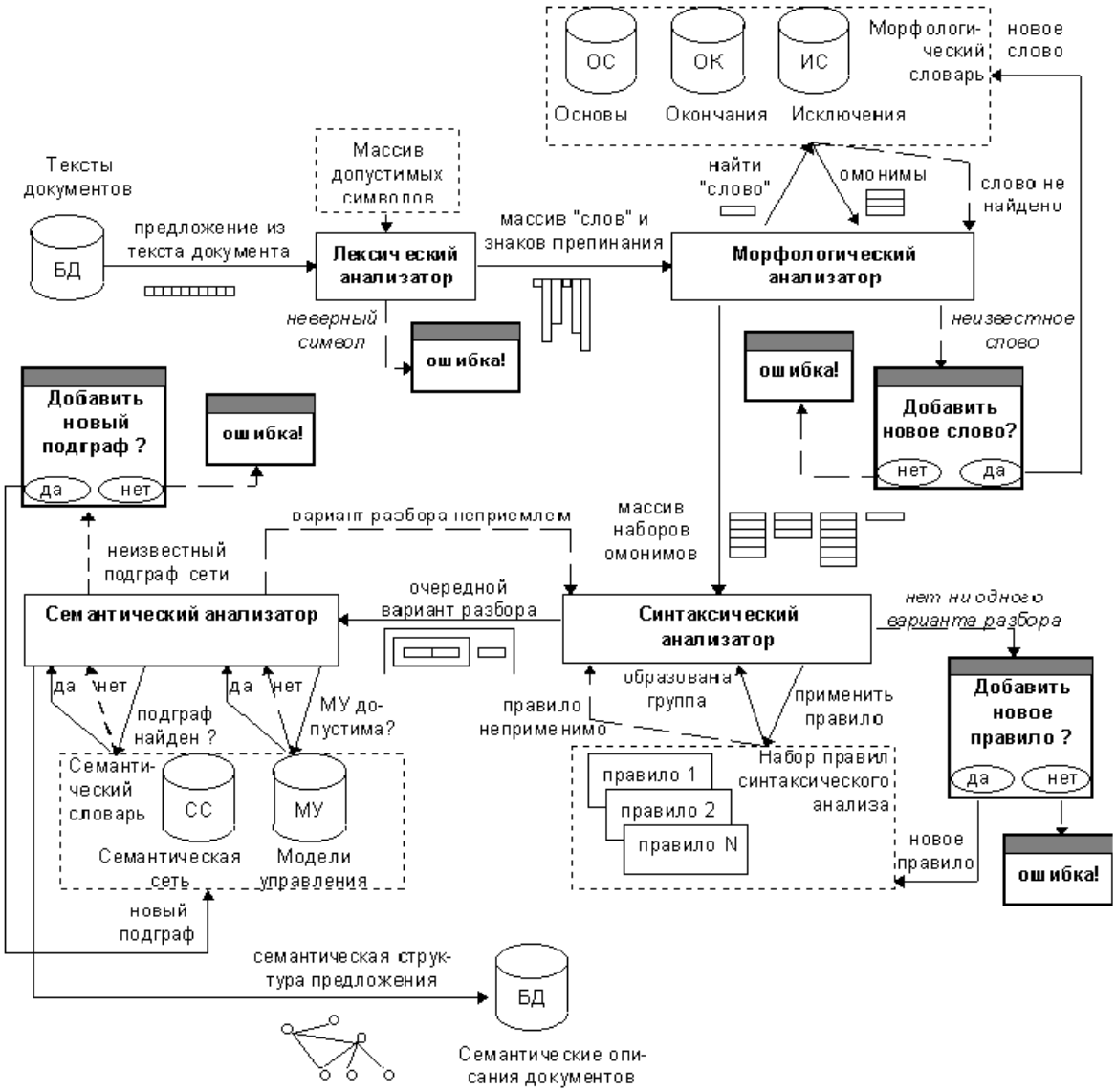

Рисунок 3.5. – Схема функціонування лінгвістичного процесора

Тому в справжній розробці зберігання вихідних і вихідних даних спрощене: вихідні тексти беруться зі звичайних текстових файлів, а результат аналізу (фрагмент семантичної мережі) записуються у двійковий файл у внутрішньому форматі семантичного аналізатора.

Висновки до 3 розділу

В результаті аналізу запропонованого методу оптимізації була розглянута доцільність використання лінгвістичного процесору, основні принципи розробки, структура й склад, опис алгоритму та структура програмного застосування.

РОЗДІЛ 4
СИНТЕЗ ОПТИМІЗОВАНОЇ ПОШУКОВОЇ СИСТЕМИ

4.1 Аналіз якості пошуку

Одна з важливих завдань, що виникають при розробці інформаційно-пошукової системи (ІПС) полягає в оцінці ефективності її роботи з порівняння з іншими подібними системами. Це необхідно як для перевірки теоретичних оцінок ефективності використовуваних методів пошуку, так і для визначення класів ситуацій, у яких найбільш доцільне використання даної пошукової системи.

Що стосується якості пошуку, те це досить суб'єктивне поняття, і оцінити його оперуючи лише формулами не представляється можливим. Оцінка якості - ідея, фундаментальна для теорії пошуку. Тому що саме завдяки оцінці якості можна говорити про застосовність або не застосовності тієї або іншої моделі й навіть обговорювати їхні теоретичні аспекти.

Оцінка якості пошуку залежить від наступних факторів:

- тематика пошуку, точність, особливості ІПС;
- психологічні, ергономічні й інформаційні фактори;
- вибір градацій релевантності, відмінність довжин вибірок - відповідей ІПС, неприступність деяких документів.

Дані фактори були класифіковані по об'єкту, який зазнав їхньому впливу. У першу групу ввійшли фактори, що впливають на сукупність документів, одержувану в результаті пошуку. У другу - фактори, що впливають на обумовлені користувачем величини релевантності (відповідності) документів. До третьої групи були віднесені фактори, що впливають на результати обчислень параметрів ефективності.

До першої групи факторів ставляться: тематика пошуку, точність формулювання пошукового запиту й особливості пошукових систем. Вплив тематики інформаційного пошуку на одержувані результати викликано декількома причинами. По-перше, залежністю від неї загальної кількості

релевантних ресурсів. По-друге, значимістю цих ресурсів для ІПС. По-третє, особливостями шуканих документів.

Популярність або специфічність, широта або вузькість тематики, зв'язана як з кількістю користувачів, так і з кількістю документів - джерел інформації. Для популярної тематики характерний надлишок інформації й труднощі вибору з багатьох альтернатив, а для вузької - відсутність або недолік інформації.

Кращі ІПС Internet містять відомості не більш ніж про 10-20 з декількох мільярдів розміщених у мережі документів. У їхнє число звичайно входять найбільш популярні інформаційні ресурси. Інша інформація не може бути знайдена через обрану ІПС. Спеціалізовані, тематичні ІПС містять відомості в основному про документи по деякій обраній тематиці. У цьому випадку гарні результати пошуку спостерігаються при збігу теми запиту з тематикою ІПС і погані - при розбіжності.

До особливостей документів, які впливають на результати пошуку, ставиться тип і стиль текстів. Відмінність властивостей типів текстів очевидно, наприклад, при порівнянні великої статті й новинного повідомлення. Різним видам документів властивий суттєво різний вид розподілу частот уживання термінів. Важлива й форма розміщення документів. Так пошуковий образ великого тексту суттєво відрізняється від образів його підрозділів. Також поважний вигляд організації зв`язності текстових документів. Існує значима відмінність між трьома класами текстів: науковими, популярними (для не фахівців) і художніми або публіцистичними текстами [13].

Другим фактором, що впливають на результати пошуку, є точність формулювання пошукового запиту. Звичайно вона недостатньо велика, що викликане явищем "язикового бар'єра" - відмінністю термінологій, використовуваних користувачем і авторами документів. Якщо шукана інформація існує в мережі й відома даної ІПС, то залежно від вірності формулювання запиту, можливі кілька класів ситуацій.

Коли запит точний, пошук буде успішним. Коли запит поганий і досить далекий від інформаційної потреби, те релевантні документи не будуть

знайдені. Найчастіше виникає проміжна ситуація, коли запит близький до потреби, але не цілком точно її описує. Тоді, можливо, частина отриманих документів буде релевантна й відповідна корекція пошукового запиту дозволить одержати гарні результати.

Третім фактором, що впливають на результати пошуку, є особливості використовуваних пошукових систем. Сюди ставляться різний вплив на результати пошуку алгоритмів їх роботи, особливостей інформаційно-пошукових мов (ІПМ), механізму індексування та ін. Для адекватного виконання пошукового запиту, він повинен бути записаний на відповідній ІПМ.

До другої групи ставляться фактори, що впливають на обумовлені користувачем оцінки документів. Ця група складається із психологічних, ергономічних і інформаційних факторів. З них найбільш значимими, є явище зміни інформаційної потреби в ході пошуку, а також стомлення користувача. Вплив цих факторів підсилюється при великій кількості використовуваних ІПС.

Розгляду підлягає явище зміни інформаційної потреби в ході пошуку. Релевантність документа - це його відповідність інформаційної потреби користувача.

Звичайно вважається, що в процесі вивчення результатів пошуку ця потреба ніяк не змінюється. Тоді значення релевантності документів не повинні залежати від послідовності їх вивчення. Однак, практика показує, що це припущення виконується далеко не завжди. Зокрема , деякі користувачі по-різному оцінюють однакові документи, що перебувають у різних частинах досліджуваної вибірки. Аналіз причин такої невідповідності показав, що основною його причиною є зміна інформаційної потреби користувача. Інші причини: дискретність шкали оцінок і неуважність користувача звичайно вносять суттєво менші викривлення.

Варто розрізняти користувача пошукової системи й експерта, що виконує оцінку якості пошуку. Коли запит точний, уже перші кілька документів задовольняють інформаційну потребу й користувач припиняє сеанс пошуку. При оцінці якості пошуку передбачається, що виконуючий її експерт розглядає

всі отримані документи. Однак, певні на основі оцінок експерта величини якості пошуку будуть залежати й від релевантності тих документів, які користувач уже не стане розглядати. Таким чином, ця експертна оцінка релевантності буде містити систематичну помилку.

Інформаційна потреба може являти собою комплекс питань, і розглянуті документи можуть містити відповіді на якусь частину з них. З іншого боку, одержання користувачем нової інформації може привести до переформулювання або виникнення нових питань. Так відбувається деяка зміна інформаційної потреби. Ця зміна носить суб'єктивний характер і залежить від знань користувача, його навченості, здатності розбиратися в новому матеріалі. Облік настільки суб'єктивних факторів є досить складним завданням, що далеко виходить за межі даної роботи.

Після зміни інформаційної потреби, відповідно зміниться область релевантних, з погляду користувача документів. Якщо експерт буде ігнорувати ця зміна і як і раніше оцінювати релевантність документів первісної потреби, то одержувані значення релевантності будуть мати більшу погрішність.

У ряді випадків відбувається інший ефект, який називається дрейфом інформаційної потреби. Він полягає в тому, що користувач високо оцінює документи, які не ставляться до шуканих, але є "цікавими" з його погляду . Часто якщо перший такий документ ще оцінюється виходячи з первісної потреби, те наступні - на основі близькості документа з новою "цікавою" темою. Таке явище часте спостерігається коли запит виявився далекий від вихідної потреби.

У всіх цих випадках зміна виражалася в зниженні інтересу до первісної потреби й появі інтересу до яких-небудь інших. Експерт, що оцінює релевантність, також підданий цьому, що вносить більші викривлення у вимірювані значення. Перспективним шляхом досліджень є моделювання процесу зміни потреби й підстроювання під нього створюваної вибірки документів. Значимість факторів зміни потреби в першу чергу визначається, очевидно, психологічними характеристиками особистості користувача, рівнем

його мотивації й т.д.. Вплив даного фактора зростає при збільшенні кількості документів або коли процес пошуку був перерваний на який-небудь проміжок часу.

Інший важливий фактор, що впливає на значення релевантності, полягає в стомленні користувача при вивченні великої кількості документів. При цьому відбувається зниження уваги, падінню інтересу, погіршується розуміння складних документів. Для зменшення впливу цього фактора бажане обмежити кількість досліджуваних документів, доцільно чергувати документи різної спрямованості, але близького стилю, створити мікропаузи й застосовувати психологічні приймання втримання уваги.

Вплив цих факторів підсилюється при збільшенні кількості досліджуваних ІПС. Крім загального зниження точності результатів, зростає значення послідовності вивчення документів. При вивченні результатів інформаційного пошуку необхідно чергувати документи, отримані від різних пошукових систем. Коли число ІПС більше п'яти-шести, у кожному сеансі пошуку потрібно працювати тільки із частиною з них. Таким чином, кожний сеанс дасть можливість побудувати часткове впорядкування їх ефективності. Потім необхідно буде об'єднати результати всіх сеансів.

До третьої групи ставляться фактори, що впливають на результати обчислень параметрів ефективності пошуку. До цих факторів належать: вибір градацій релевантності, відмінність довжин вибірок - відповідей ІПС, неприступність деяких документів та ін. Значимість факторів цієї групи визначається вибором характеристик ефективності інформаційного пошуку.

Коли довжини вибірок, формованих ІПС, не збігаються, можливі два способи визначення шуканих характеристик. Перший полягає в розрахунках параметрів згідно з формулами. Другий припускає скорочення всіх вибірок до розміру найменшої й розгляд тільки документів з урізаних вибірок. Скорочення довжини вибірки може, зокрема , збільшити впорядкованість і унеможливити визначення номера першого релевантного документа. Становить інтерес

порівняльний аналіз пошукових систем по характеристиках, отриманих для декількох довжин вибірок.

Часто деякі документи, що втримуються у відповідях ІПС, виявляються недоступними. З одного боку, можна просто вважати їх нерелевантними. Однак, документи можуть автоматично перевірятися на доступність при використанні мета-пошукових систем. Відповідно, другий варіант полягає у виключенні цих документів з вибірки, що приводить до зменшення її довжини.

При порівнянні ефективності різних ІПС необхідно одержати представницький набір пошукових сеансів різної тематики, у тому числі: художньої, популярної, наукової. Сеанси пошуку слід розділити на три групи по близькості запиту й інформаційної потреби. Усереднення отриманих значень якості пошуку має сенс тільки роздільно для цих груп. Коли досліджується більш п'яти-шести пошукових систем, у кожному сеансі пошуку потрібно працювати тільки із частиною з них. Необхідно ретельно визначити адекватні характеристики якості результатів пошуку, побудувати правила для обчислення їх значень у різних ситуаціях [13].

4.2 Релевантність

Поняття релевантності не є специфічним для систем інформаційного пошуку. Воно з'явилося з філософських теорій, що пояснюють відносний зв'язок між джерелами інформації, і вивчається багатьма напрямками науки.

Цікаво, що в перші роки розвитку області інформаційного пошуку розглядався ряд альтернатив поняттю релевантність, наприклад, популярне в експертних системах поняття невизначеність. Досить імовірно, що якби вибір припав на яку-небудь із альтернатив, то розвиток систем інформаційного пошуку пішов би зовсім по іншому шляхові.

Обговорення поняття релевантності в контексті інформаційно-пошукових систем ведеться вже близько півстоліття, але його детального загальноприйнятого визначення усе ще немає.

Докладне обговорення альтернативних підходів до концептуалізації релевантності виходить за рамки цього огляду. Тому варто обмежитися лише описом семантичної різниці між декількома найбільш популярними видами релевантності й обговорити ступені релевантності.

Для класифікації типів релевантності можна скористатися одним з каркасів, використовуваних для концептуалізації відносини релевантності. Цей каркас має три розмірності:

Інформаційна потреба (Infneed).

Виділяється 4 вистави інформаційної потреби**:**

- реальна потреба (*RIN*) - неусвідомлена дійсна інформаційна потреба користувача (наприклад, пошук якоїсь нової інформації дослідником, про яку він доладно нічого не знає**;**

- усвідомлена потреба (*PIN*) - те, як користувач розуміє варту перед ним неусвідомлену проблему**;**

- виражена потреба (*EIN*) - те, як користувач описує свою потребу засобами природної мови**;**

- формалізована потреба (*FIN*) - вистава EIN засобами мови запитів пошукової системи.

Ці вистави можна впорядкувати по ступеню їх потенційної відмінності від інформаційної потреби:

$$\boldsymbol{FIN} \le \boldsymbol{EIN} \le \boldsymbol{PIN} \le \boldsymbol{RIN} \qquad (4.1)$$

Інформаційні ресурси (Infres). Виділяється 4 типу інформаційних ресурсів, які можуть бути доступні користувачеві в процесі пошуку:

- безліч документів (DS) - набір документів, які разом задовольняють потреба користувача**;**

- документ (D) - повний інформаційний ресурс, посилання на який представляється користувачеві в результаті пошуку**;**

- метаінформація (MD) - структурована інформація про документ, така як, наприклад, бібліографічна інформація, характеристика якості документа або відкликання інших користувачів**;**

- сурогат (S) - вистава документа у вигляді заголовка, автора, анотації й т.п..

По потенційній можливості надання користувачеві необхідної інформації ці сутності також можна впорядкувати:

$$\boldsymbol{S} \leq \boldsymbol{MD} \leq \boldsymbol{D} \leq \boldsymbol{DS} \qquad (4.2)$$

Контекст використання інформації (Infctx).

Цей контекст концептуалізується за допомогою трьох компонентів:

- тематика (To) - область інтересів користувача**;**

- завдання (Ta) - процес або завдання, для розв'язку якої користувач ініціював пошук**;**

- атрибути користувача (UA) - опис характеристик користувача, таких як його знання в цій тематиці або час, протягом якого він прагне знайти відповідь**.**

Оскільки кожна із цих компонентів важлива, те можна розглядати будь-які їхні комбінації.

Таким чином, відношення релевантності можна описати як:

$$\boldsymbol{Relevance\ (InfNeed,\ InfRes,\ InfCtx)} \qquad (4.3)$$

Важливо, що поняття релевантності не статично, а може змінюватися в часі через зміну якого-небудь його компонента, що характеризує. Наприклад, прочитання документа змінює не тільки знання користувача, але може викликати й зміна формалізованої (або навіть усвідомленої) потреби.

Слід зазначити, що описаний каркас не є ідеальним або єдино можливим.

4.2.1 Види релевантності

Опираючись на каркас, описаний вище, є можливим відносно чітко визначити різницю між деякими популярними видами поняття релевантність, використовуваними при оцінці систем текстового пошуку.

Когнітивна релевантність або *пертинентність* - це відношення, що характеризують відповідність реальної потреби користувача й інформації з документа, тобто *Relevance(RIN)*. Це "ідеальна" релевантність, усі інші види релевантності характеризують її наближення з різних точок зору.

Тематична або предметна *релевантність* - це відношення, що характеризує близькість тематик потреби й ресурсу, тобто *Relevance(-,-,To)*. Вона звичайно використовується, коли оцінка проводиться на рівні обробки.

Ситуаційна релевантність або *корисність* - це релевантність ресурсу в контексті розв'язуваної користувачем завдання, тобто *Relevance(-,-,Ta)*. або Relevance(-,-,To+Ta). Наприклад, корисність при ухваленні рішення, відповідність інформації розв'язуваній проблемі й т.п. Цей тип релевантності звичайно мається на увазі при оцінці на рівні виходу.

Що спонукує або емоційна *релевантність* - це релевантність ресурсу в контексті поточної ситуації, тобто *Relevance(-,-,UA)*. Вона звичайно використовується при оцінці на рівні застосування.

Дуже важливим поняттям є також системна або алгоритмічна релевантність - це оцінка релевантністи між формалізованою інформаційною потребою й документом, дана пошуковою системою. При цьому може також ураховуватися й вистава системи про компоненти, що характеризують контекст використання інформації.

Існує ще два види релевантності - *організаційна* й *соціальна*. Вони звичайно використовуються при оцінці на соціальному рівні. Обоє цих виду релевантності не можуть бути описані в рамках використовуваного каркаса, тому що ставляться до завдання оцінки в інших контекстах, а саме в контексті організації й контексті суспільства відповідно. Оскільки обоє цих контексту перебувають поза рамками цього огляду, то тут варто лише відзначити, що вони

є аналогами, що спонукує релевантності й залежать від очікувань організації або суспільства.

4.2.2 Ступені релевантності

Відносність поняття релевантність є прямим наслідком його походження з філософських обговорень релятивізму. Це обумовлює появу концепції різних ступенів релевантності, що характеризують часткову релевантність документа запиту.

Концептуально ступінь релевантності можна вимірювати будь-яким речовинним числом від 0 до 1, але оскільки людина не здатна чітко характеризувати ступінь релевантності, те її часто оцінюють за допомогою k-значної шкали.

На практиці в більшості досліджень по оцінці систем пошуку використовуються бінарні оцінки ***(k=2)***, хоча є ряд випадків, коли більша кількість градацій корисно.

Відомо, що ступінь релевантності сильно залежить від завдання ***To*** і характеристик користувача ***UA***. Так, наприклад, було виявлено, що чим більше користувач знає про те, що він шукає, тем менше документів, які він уважає частково релевантними [13].

4.3 Застосована модель ранжування

У якості математичної моделі інформаційного пошуку для ранжирування результатів видачі документів була обрана булева модель.

У загальному випадку критерієм релевантності документа запиту є істинність булевого вираження, заданого в запиті.

Розгляду підлягає приклад пошуку "по тексту".

Матриця документ-термін C(d,t) (таблиця 4.1) показує, які зустрічаються слова й у яких документах, де di - документ, ti - терм.

Таблиця 4.1 – Матриця документ-термін C(d>t)

C(d,t)	*t1=a*	*t2=b*	*t3=c*
d1	**1**	**0**	**0**
d2	**1**	**1**	**0**
d3	**1**	**0**	**1**
d4	**0**	**1**	**0**
d5	**1**	**1**	**1**

Запит: *q = a* І *(b* АБО (НІ *c))*

a-> 1,1,1,0,1
b-> 0,1,0,1,1
НІ *c* ->1,1,0,1,0

і 1,1,0,0,1

або 1,1,0,1,1

Результат: **d1, d2, d5**

Одним з безсумнівних переваг є простота її реалізації. Головним недоліком уважається відсутність можливості ранжирування знайдених документів по ступеню релевантності, оскільки відсутні критерії її оцінки.

Для забезпечення можливості ранжирування видаваних користувачеві документів, може ускладнюватися булева модель пошуку. Запропоновано кілька варіантів так званих розширених булевих моделей. У цих моделях уводяться спеціальні узагальнення булевих операторів, що дозволяють додати підвищена вага документам, у точності задовольняючих булевому вираженню запиту, і знижена вага - усім іншим документам [4].

4.4 Використання лексичних функцій для перифразування

У самому загальному виді можна сказати, що лексичні функції - це тривіальні змісти, словесне вираження яких у тексті залежить від того, при якому конкретнім слові цей зміст виражається. Для деяких фрагментів лексичної системи мови розроблені лексичною семантикою правила виду: "При слові X зміст f1 виражається словом X′, при слові Y зміст f1 виражається

словом Y′" мають велику передбачену силу. Одним з таких фрагментів є клас параметричних слів. Під параметричними словами ми розуміємо імена іменники зі значенням параметра, що допускає числове значення, наприклад: висота, місткість, обсяг; тривалість, вік; потужність, сила, маса, тиск, магнітуда; народжуваність, смертність; ціна, вартість, зарплата, виторг; ентропія, рівень, коефіцієнт, індекс і т.д. [10].

Правила лексичної семантики, що описують функціонування слів цього класу, у російській мові носять особливо строгий характер. Це пов'язане з характерною рисою мови: хоча параметри є універсальним типом предикатів, прототипічними представниками даного класу в російській мові є не дієслова, а іменники. Дієслів з відповідними значеннями, таких як коштувати, важити, тривати, уміщати, дуже мало. Навіть для вираження такого тривіального параметра, як 'висота', спеціалізованого дієслова немає (у відмінність, наприклад, від англійського: The Pisa tower rises 56 meters). Тому природнім способом приписування якому-небудь об'єкту певного параметра є конструкція з параметричного іменника й дієслова (Пізанська вежа досягає у висоту 55 метрів).

Відносини між іменником і дієсловом в твердій формі з'являються як відносини назви ситуації й допоміжного дієслова, що служить для вираження категоріальних значень - виду, часу і т.д. При більш уважному розгляді виявляється, що невільні комбінації виду " дієслово-зв'язування +іменник" діляться на групи, об'єднані загальним елементом змісту, який і є значення лексичної функції.

Апарат лексичних функцій лежить в основі опції перифразування, реалізованої в системі ЕТАП-3. Перифразування дозволяє, опираючись на записані в словникових статтях слів значення лексичних функцій, побудувати деяка кількість пропозицій, синонімічних або квазисинонімічних заданому, тобто пропозицій, що мають із ним однакову структуру на глибинному семантичному рівні. У системі ЕТАП для опису невільних словосполучень, загалом кажучи, використовується кілька десятків лексичних функцій. Вони

дозволяють із пропозиції «Чистка матрицы стоит десять гривен», одержати наступний набір пропозицій:

Чистка матрицы имеет стоимость десяти гривен;

Чистка матрицы достигает стоимости десять гривен;

Чистка матрицы имеет цену на десять гривен;

Стоимость чистки матрицы составляет десять гривен;

Стоимость чистки матрицы достигает десяти гривен;

Стоимость чистки матрицы равняется десяти гривнам;

Цена чистки матрицы составляет десять гривен.

За допомогою тих же лексичних функцій можна, подавши на вхід кожне із пропозицій, що вийшли, знову одержати весь набір перифраз.

Опція перифразування була пристосована для розв'язку пошукових завдань у такий спосіб: з іменного словосполучення, у вершині якого перебуває параметричне слово, можна одержати ряд неповних речень, що містять усі елементи, крім чисельного значення параметра.

Уведення: глубина Марианской впадины. Перифрази, генеровані системою перифразування ЕТАП-3:

Глубина Марианской впадины равна;

Глубина Марианской впадины составляет;

Глубина Марианской впадины достигает;

Глубина Марианской впадины равняется;

Марианская впадина имеет в глубину;

Марианская впадина достигает в глубину;

Марианская впадина имеет глубину;

Марианская впадина достигает глубины.

4.5 Алгоритмічна організації перифразування

Торкнемося лише того фрагмента перифразування, який ставиться до нашого завдання. Наведений вище кущ перифраз будується на основі

інформації про лексичні функції, що втримується в словниковій статті параметричного слова глибина, і деяких універсальних правил перифразування.

Для нашого завдання в слові глибина інтерес представляють записи, що ставляться тільки до трьом функціям:

OPER1:ИМЕТЬ/ДОСТИГАТЬ

FUNC2:СОСТАВЛЯТЬ1/ДОСТИГАТЬ/РАВНЯТЬСЯ1

LABOR1-2:ИМЕТЬ<В1>/ДОСТИГАТЬ<В1>

Із усього різноманіття лексичних функцій, що обслуговують слово глибина ці три з'єднують ім'я параметра з дієсловом таким чином, щоб вийшов опис ситуації, коли що-небудь має певну глибину. Функція OPER1 дозволяє позначити ситуацію так, що параметр виступає при відповіднім дієслові першим доповненням, наприклад: Гора Котопакси має висоту майже 6 км. Функція FUNC2 дозволяє позначити ситуацію так, щоб параметр був підметом при функціональнім дієслові: Висота гори Котопакси становить майже 6 км. Функція LABOR1-2 дозволяє так позначити ситуацію, що параметр займає місце другого доповнення при функціональнім дієслові, а місця підлягаючого й присудка займають перший і другий актанти відповідно: Гора Котопакси достигает 5 870 м в высоту.

З універсальних правил перифразування (їх у системі ЕТАП кілька десятків) задіяні лише три двосторонні правила:

OPER1 + X ↔ FUNC2 + X (иметь глубину <--> глубина составляет)

OPER1 + X ↔ LABOR1-2 + X (иметь глубину <--> иметь в глубину)

FUNC2 + X ↔ LABOR1-2 + X (глубина составляет <--> иметь в глубину)

Якщо яка-небудь лексична функція має кілька значень (у нашому прикладі всі три функції представлені альтернативними значеннями), то система перифразування будує пропозиції по черзі з усіма значеннями. Те самий дієслово може бути значенням різних лексичних функцій (у нашому прикладі: досягати висоти, висота досягає, досягати у висоту). У різних параметричних слів значення цих трьох лексичних функцій часто збігаються, але зустрічаються

й помітні відмінності. Наприклад, для слова «мощность» функція OPER1 має три значення:

OPER1:ИМЕТЬ/ДОСТИГАТЬ/РАЗВИВАТЬ, а функція LABOR1-2 для цього слова не існує.

Наведений вище кущ перифраз складається з 8 пропозицій. Система перифразування побудує цей кущ цілком, якщо їй на вхід подадуть або іменну групу (як у наведеному вище прикладі), або будь-яка пропозиція із цього куща.

Відзначимо, що для розв'язку нашого завдання треба було встановлення окремого режиму роботи системи ЕТАП. Справа в тому, що всі правила перифразування в ЕТАПІ настроєні на роботу з повними фразами, а фрази наведеного вище куща повними не є. Тому при цьому режимі роботи системи після одержання синтаксичної структури вхідного (неповного) пропозиції проводиться добудування цієї пропозиції до повного. Наприклад, якщо на вході була іменна група глибина Марианской западини, те добудовуємо цю пропозицію до глибина Марианской западини рівняється чомусь. Дієслово рівняється - чергове значення лексичної функції FUNC2 для абсолютної більшості параметричних слів, а іменник щось виконує роль тимчасового доповнення, необхідного при перефразовуванні. Після синтезу чергової перифрази цей тимчасовий іменник стирається. Якщо на вхід ЕТАПА подається одне із пропозицій куща, де дієслово присутній, то синтаксична структура поповнюється тільки тимчасовим доповненням.

Згадаємо ще одну проблему, яка виникає, коли на вхід подається іменна група. Наприклад, для запиту «водоизмещение ’Титаника’» наша система у даний момент побудує запити: Водоизмещение ‘Титаника’ составляет, ‘Титаник’ имеет водоизмещение, у яких дієслова коштують у теперішньому часі. Такі запити не можуть поліпшити пошук, оскільки форма точного запиту не припускає в якості результату словоформи, що відрізняються від заданих наборами граматичних характеристик, а в текстах про 'Титаникетитаните' ці дієслова, швидше за все, ужиті в минулому часі. Подібна ситуація може виникнути й з характеристикою виду дієслова (сов/несов). Розв'язок цієї

проблеми міг би полягати в породженні всіх перифраз, де дієслова мають різні характеристики часу й виду. Алгоритмічно це зробити не складно, але число перифраз при цьому помітно зросте.

У той же час згадана проблема зникає, якщо на вхід подається запит, що містить дієслово в необхідній для даного запиту формі:

Водоизмещение ‘Титаника’ составляло;

У цьому випадку система перифразування збереже для всіх перифраз характеристики дієслова, що зробило на вхід.

4.6 Практичний аналіз перефразування

На основі вихідного списку з 100 параметричних слів було складено близько 120 осмислених коротких запитів (параметр + носій параметра). Саме до таких запитів, наскільки можна судити по статистиці більшості ПС, найчастіше прибігають користувачі. За допомогою системи ЕТАП-3 кожний запит перетворювався в блок перифраз. У ході експерименту, перифрази вводяться по одній у пошукові системи. Оскільки нас цікавила принципова можливість поліпшити пошук, автоматично розширюючи запит, час виконання запиту й час обробки запиту системою ЕТАП-3 не враховувалося, так само як і дата проведення експерименту й завантаження пошукових серверів. Оскільки система перифразування видає цілісні структури, що не виходять за межі одного пропозиції, що й не передбачають розривів і пропусків, для тестування була обрана форма точного запиту. Застосування логічного оператора "АБО" усередині точного запиту мовами пошукових запитів не підтримується. Уживання диз'юнкції при неточному запиті приводить до того, що перебувають не сайти із цілісною структурою, а сайти, що просто містять задані слова, можливо, не зв'язані синтаксично. Тому група слів, що виражає чисельне значення й залежна від дієслова, може ставитися до в зовсім іншому об'єкту або до іншого параметра. Частково знімає цю проблему опція пошуку із пропуском заданого кількості слів. Можна собі представити неточний запит виду

водотоннажність Титаника /+1 становити. Однак і він не дає стовідсоткової точності. На першу ж сторінку чомусь попадає такий фрагмент:

Через два часа тридцать пять минут после катастрофы крен "Титаника" составлял почти 90 градусов.... Длина парома "Геральд оф Фри Энтерпрай" составляла 132 метра, водоизмещение - 7951 тонна. Он являлся составной частью флота, управляемого компанией...

Очевидно, ці опції поки ще не завжди коректно обробляються пошуковими машинами. Якщо ускладнити запит, наситивши його "лінгвістичними" маркерами (наприклад "Гора Котопакси" /+1 достигает /+3 "в высоту"), частка невідповідних відповідей ще зросте.

Уживання точної форми запиту привело до того, що відмінності в роботі двох згадані пошукові систем практично нівелювалися, за винятком відмінностей у складі баз документів (наприклад, "Яндекс" індексує більше особистих блогів, чому Google). Ці відмінності виявилися настільки незначними, що ми порахували можливим ними зневажити.

Ціль полягала в тому, щоб визначити, наскільки системи перифразування підвищує ефективності пошуку. Для цього був розроблений наступний протокол оцінки ефективності. Релевантним зізнається такий результат, коли шукана чисельна інформація втримується безпосередньо в снипеті, пропонованому пошуковою машиною. Показником ефективності пошуку вважається кількість релевантних результатів на першій сторінці (тобто кількість релевантних результатів серед перших десяти), виражене у відсотках. Якщо кількість знайдених документів менше 10, то воно й ухвалюється за 100%. У якості контрольного рівня ефективності пошуку був прийнятий рівень ефективності пошуку по вихідних запитах - іменним групам, що подаються на вхід системи перифразування ЭТАПА-3. Ці запити також оформлялися як точні - це обмеження дає можливість оцінити чистий ефект використання перифраз.

У результаті цих перифраз одержуємо дані по кожному запиту:

Таблиця 4.2 – Результати перефразовування даних по запиту

Запит	Знайдено сторінок	Знайдено сайтів	Кіл-ть релевантних документів на 1 сторінці
“твердость алмаза”	2044	791	3/10 (30%)
“твердость алмаза равна”	42	18	10/10 (100%)
“твердость алмаза составляет”	18	6	3/6 (50%)
“твердость алмаза достигает”	0	0	0
“твердость алмаза равняется”	0	0	0
“алмаз имеет твёрдость”	79	35	9/10 (90%)
“алмаз достигает вёрдости”	0	0	0

Перший рядок у даних по кожному запиту буде представляти ефективність пошуку по неопрацьованих словосполученнях, а всі інші - по оброблених перифразах. Щоб оцінити середня зміна результату, можна поєднати середнє значення ефективності неопрацьованих запитів і середнє значення по всіх перифразах разом. Однак такий підхід видасться нам невірним. Уявимо собі, як могла б працювати пошукова машина, що вміє генерувати перифрази. Вона одержує на вході запит, створює перифрази, а потім обробляє кожну з них як точний запит. Якісь перифрази приносять результат, якісь - немає. На екран у так випадок виводи сума знайдених по вазі перефраз документ, а перефрази, що не спрацювали, нічого не додають, але і не покращують Тому ми розраховували підвищення ефективності пошуку для кожного запиту окремо, і вже для цих результатів потім розраховувався середній показник, що характеризує загальну зміну ефективності. Так, для наведеного вище запиту середній показник збільшення ефективності складе:

((100%-30%)+(50%-30%)+(90%-30%))/3=50% або 0,5.

Бувають випадки, коли неопрацьований запит приносить якась кількість релевантних відповідей, а перифрази не приносять нічого. Наприклад, "абсолютный минимум температуры на Земле". Це пов'язане з тим, що інформація із цього питання дуже часто представлена передруком статті з

Великої радянської енциклопедії, де використана конструкція з нульовим зв'язуванням замість дієслова. У цьому й інших схожих випадках ми визнавали результат використання системи перифразування негативним: якщо результати пошуку по неопрацьованому запиту були релевантні в 10 випадках з 10, то ефективність пошуку з перифразуванням рівна -100% або -1, якщо в 6 випадках з 10 - -60% або -0,6 і т.д..

Розподіл показників зміни ефективності пошуку при використанні системи перифразування презентовано на гістограмі.

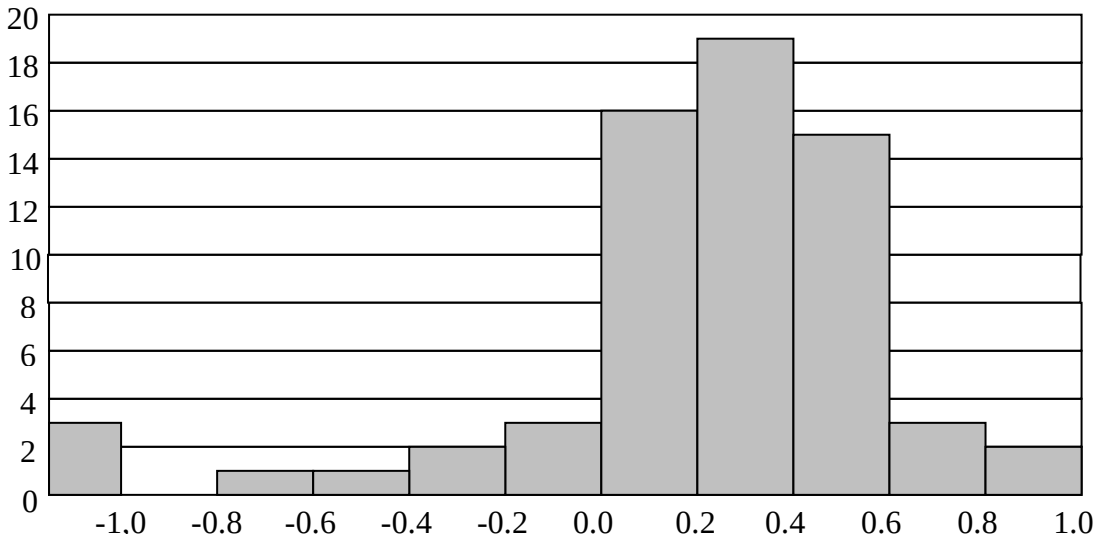

Рисунок 4.1 – Зміна ефективності пошуку при використанні запитів на основі перифраз

На графіку видно, що в середньому точність пошуку підвищується на 24%. Цікаво, що якщо розділити параметричні слова по тематичних групах, те цей середній показник буде варіюватися. Якщо дивитися дані по "географічних" запитах (параметри географічних об'єктів, як природних, так і культурних, наприклад, ширина ріки Амазонки в середньому плині або чисельність населення Києва), то підвищення ефективності пошуку складе в середньому 18,5%, а в групі даних по фізичними запитам (приклади запитів: молярна маса міді, сила ваги на Марсу) цей показник буде рівний 27%. Результати пошуку по геометричних об'єктах (приклади запитів: площа поверхні сфери, обсяг конуса) виявилися несподіваними. Що реалізує значення лексичної функції FUNC2

дієслово рівнятися, а також додана в пошукове перифразування конструкція з коротким прикметником рівний, відрізняються від інших засобів тим, що регулярно приєднують у якості першого доповнення формулу або її словесний опис: Обсяг конуса рівний однієї третини добутку підстави на висоту; Обсяг конуса рівняється однієї третини обсягу циліндра з тими ж підставою й висотою. Показник ефективності використання перифраз для пошуку інформації такого роду становить 13%. Однак ЕТАП-3 можна настроїти на пошук подібної інформації, додавши в систему перифразування дієслова характеризувати, вимірятися й, можливо, деякі інші. Окремо підкреслимо, що ці дієслова не є значеннями лексичних функцій параметричних слів. Точно так само не описуються лексичними функціями конструкції типу площа восьмикутника може бути обчислена як.... Інформацію про те, що формули можуть уводитися подібними пропозиціями ми черпаємо з екстралінгвістичних фактів. Відповідно, у рамках системи ЕТАП можна побудувати ряд правил, що дозволяють одержувати подібні пропозиції, однак ці правила, швидше за все, будуть украй неповними: при нічим не обмеженої сполучуваності слів практично неможливо описати всі можливі пропозиції й тем паче вгадати пропозиції, реально присутні в Інтернет-документах.

4.7 Переклад запитів

Оскільки однієї з опцій процесора ЕТАП є автоматичний переклад тексту, то для системи не становить праці перевести всі отримані шляхом перефразування словосполучення на англійський. Із запиту висота Пізанської вежі за допомогою ЕТАПА легко одержати набір словосполучень :

The height of the Pisa tower equals;

The height of the Pisa tower reaches;

The height of the Pisa tower is reaching;

The height of the Pisa tower amounts to;

The height of the Pisa tower attains;

The height of the Pisa tower is attaining.

Однак ефективність пошуку в цьому випадку практично не підвищується. І це знову пов'язане з особливостями ладу мови, зокрема , з тим добре відомим фактом, що в англійській мові параметри задаються прикметниками зі значенням верхнього полюса шкали (30 feet high, 6 feet tall, 25 years old), що російській мові практично не властиво (за винятком відомих рідких прикладів типу як велика ймовірність п). У випадку параметрів для носіїв мови природніше вживати прикметники: The Cupola is 55 meters high and 16 meters wide. Тому перифрази, побудовані ЕТАПОМ за допомогою існуючих на сьогоднішній день правил, незважаючи на те, що граматично вони абсолютно правильні, можуть взагалі на загал не зустрічатися серед джерел. Саме так інша справа з Пізанською вежею.

Крім того, відіграє роль загальновідомий факт обов'язковості зв'язку дієслів в англійській мові. У тих випадках, коли параметр все-таки позначається іменникам, носіям англійської мови не потрібно підбирати дієслово, яке могло б виразити граматичні категорії - досить просто використовувати дієслово be: The speed of light is 300 million metres per second. На даному рівні експериментальне перифразування не може бути використане для перекладу запитів. Однак саме в силу того, що в англійському різниця між формою запиту й формою відповіді настільки помітна, це поле діяльності представляється, що досить інтригують. Щоб рухатися в цьому напрямку, необхідно розширювати як правила перифразування, так і інструменти, що дозволяють переводити глибинні англійські структури пропозицій типу The Cupola is 55 meters high у глибинні російські структури пропозицій типу Висота купола становить 55 метрів. Це може стати кроком до побудови повноцінної глибинно-синтаксичної вистави в тій діючій моделі мови, яка лежить в основі лінгвістичного процесора ЕТАП-3.

4.8 Неточні запити

Дані експерименту показують, що використання перифраз підвищує точність пошуку за рахунок вибору тільки тих документів, які містять шукану інформацію, наявність якої однозначно передвіщається дієсловом, що

реалізують ту або іншу лексичну функцію. Однак вимога точного запиту приводить до того, що відсіваються й релевантні документи, у яких думка виражена небагато по-іншому: не буде знайдений документ, що містить пропозиція Гора Котопакси, висота якої становить майже 6 км… Неточний пошук по перифразі приводить до того, що конструкція розривається й по запиту тривалість життя в Голландії становить, перебувають сторінки новин, де є інформація про тривалість життя в Китаї й повідомлення про політичну ситуацію в Голландії. Однак деякі спроби такого пошуку показують, що й у цьому випадку точність пошуку по перифразах вище точності пошуку без них. Це відбувається через те, що наявність дієслова, нехай навіть і відірваного від іменного словосполучення, міняє загальну спрямованість сторінки. Наприклад, по запиту висота юки перебувають сторінки, що містять оголошення про продаж пальми якої-небудь висоти, а по запиту висота юки досягає перебувають статті з різноманітних довідників по квітництву, що містять загальну інформацію про рослину. Можна сказати, що в дію вступає стилістичний фактор, тому що дієслова, що реалізують лексичні функції OPER1, FUNC2, LABOR1-2 часто використовуються в науково-публіцистичних і наукових текстах. Це відкриває для системи перифразування певні перспективи, тим більше що в системі ЕТАП-3 передбачалося застосування стилістичних фільтрів.

Точність результатів пошуку по запитах, що припускають чисельні відповіді, може бути збільшена за рахунок використання системи перифразування. У випадку точного запиту кількість релевантних результатів збільшується на 24%. Точний запит означає, що до шуканих документів пред'являються самі тверді вимоги. У випадку менш твердих вимог, тобто при використанні нестрогого запиту, результати непередбачені. Крім варіювання форм запитів, що залежить від пошукової машини, шляхи поліпшення роботи перифразування лежать у розширенні числа використовуваних лексичних функцій і більш точного їхнього настроювання на різні типи інформації.

4.9 Опис системи

Розроблювальна пошукова системи - засіб повнотекстової індексації й наступного пошуку інформації. ПС будує глосарій зі слів, знайдених на web-сторінках, та за допомогою лінгвістичного процесору будується кущ перефраз для запиту. При пошукових запитах, вона відображає сторінку з результатами пошуку по ключових словах, ранжируваними по кількості входжень у глосарій.

Модуль індексування підтримує наступні види посилань:

- класичних посилань виду <a href..>;

- посилань із документів, що містять фрейми;

- сценаріїв клієнтської мови JavaScript;

- редірект (перенапрямок) з наступною індексацією вмісту.

При настроюванні системи, можна вказати глибину пошуку, указавши значення параметрів X(посилання) і Y(глибина), де максимальна кількість індексованих документів D = ((X * Y) + 1).

Індексуються будь-які сторінки html, як динамічні так і статичні. Також система визначає Mime-type документа.

Індексування документів проводиться з урахуванням стоп-слів, зазначених в текстовому файлі. Стоп-слова це слова, які не додаються в індекс.

Ще однієї опцією є можливість виставляння пріоритету поля title для документів у процесі ранжирування результатів пошуку.

Системою забезпечується:

- підтримка text/html, text/xml, text/plain, і image/gif;

- індексування багатомовних сайтів.

- підтримка <META NAME="robots" content="..."> і robots.txt.

ЗАГАЛЬНІ ВИСНОВКИ

У дипломній магістерській роботі на основі виконаних досліджень, відповідно до технічного завдання була вирішена науково-технічна задача по оптимізації алгоритму пошукової системи.

В результаті виконання отримані наступні наукові і практичні результати:

1 За результатом аналізу існуючих аналогів, у якості математичної моделі ранжирування результатів була обрана найоптимальніша булева модель. Згідно з моделлю був розроблений алгоритм для подальшої реалізації програмної частини комплексу.

2 На основі проведених досліджень, у програмному коді були застосовані модулі індексування сторінок і ранжирування результатів пошуку, а також допоміжний модуль включений до комплексу, який веде статистики по кількості входжень слів на web-сторінках і запитів, що вводяться користувачами пошукової системи.

3 Застосування запропонованого методу оптимізації для пошукової системи, у випадку точного запиту дозволяє збільшити кількість релевантних результатів на 24%. Точний запит означає, що до шуканих документів пред'являються самі тверді вимоги. Крім варіювання форм запитів, що залежить від пошукової машини. Також досліджено що шляхи поліпшення роботи перифразування, яке є основою запропонованого методу, лежать у розширенні числа використовуваних лексичних функцій і більш точного їхнього настроювання на різні типи інформації. Ці результати показують що використання лінгвістичного процесора для оптимізації являє приклад того, що завдання підвищення точності пошуку в Інтернеті можна добре вирішувати не тільки математичними методами.

Завдання на дипломну магістерську роботу виконане в повному обсязі.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Андреев А.М., Брик А.В., Березкин Д.В. Лінгвістичний процесор для информаційно-пошукової системи. – М.: Наука, – 2006, С. 156.
2. Апресян Ю.Д., Богуславський И.М. Лингвістичний процесор для важких інформаційних систем. – М.: Наука, – 2008, С. 280.
3. Ашманов І., Іванов А. Оптимізація сайтів у пошукових системах. – СПБ.: Питер, – 2008, С. 203.
4. Барахнін В.Б, Федотов А.М. Ресурси мережі Інтернет як об'єкт наукового дослідження, Звістки вузів. – №1. – 2008. С. 349.
5. Богуславский U. М. Лингвистический процессор и локативные обстоятельства // Вопросы языкознания № 1, – М.: Идея-Пресс. – 2005, С. 301.
6. Гасанов Е.Е. Зберігання й пошук інформації. – М.: Либроком. – 2003, С. 279.
7. Дубинский А.Г. Некоторые вопросы применения векторной модели представления документов в информационном поиске // Управляющие системы и машины. – №4. – 2008. – С. 143.
8. Жолковский А.К., Мельчук И.А. До побудови діючої моделі мови «Зміст ↔ Текст» // Машиний переклад и прикладна лінгвістика. №11. – 2006. С. 189.
9. Иомдин Л.Л. Автоматическая обработка текстов на естественном языке. Модель согласования. – М.: Наука, – 2007. – С. 290.
10. Карпова Г.Д., Пирогова Ю.К., Кобзарева Т.Ю., Микаэлян Е.В. Компьютерный синтаксический анализ: описание моделей и направлений разработок. // Итоги науки и техники (серия “Вычислительные науки”). Т.6. – М.: ВИНИТИ, 2005, – С. 290.
11. Когаловский М. Р. Перспективные технологии информационных систем. – М.: ДМК Пресс; М.: Компания АйТи, 2007. С. 234.
12. Кулагина О.С. Дослідження з машинного перекладу. – М.: Наука. – 2005, С. 320.

13. Кулагина О.С. Машинний переклад. Сучасний стан. // Семіотіка і інформатика, – М.: «Вильямс» Вип. 29, – 2008, С. 287.
14. Кураленок И.Е., Некрестьянов И.С. Оценка систем текстового поиска. Программирование, – 2007. – С. 346.
15. Ландэ Д. В., Снарский А. А., Безсуднов И. В. Интернетика: Навигация в сложных сетях: модели и алгоритмы. – М.: Либроком, 2009. – С. 264.
16. Мельчук И.А. Опыт теории лингвистических моделей "Смысл Текст". – М.: Наука, 2006. – С. 314.
17. Митюшин Л.Г. Алгоритм синтаксического анализа в системе ЭТАП. – М.: Наука, – 2006. – С. 236.
18. Некрестьянов И. Как оценить качество поиска в Интернет. СПГУ, 2009. – С. 45-46.
19. Нюссбаум Х., Интрона Л.Д. Формирование Сети: почему важна политика поисковых машин // Интернет в общественной жизни. – М.: Идея-Пресс, – 2006. – С. 12-36.
20. Олифер В.Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, протоколы. Учебник для вузов. П.: «Питер», – 2007. – С. 536.
21. Перцов Н.В., Санников В.З., Цинман Л.Л. Лингвистическое обеспечение системы ЭТАП, – М.: Наука, – 2009. – С. 295.
22. Рувимович М.К. Перспективные технологии информационного поиска в Интернете. – М.: ДМК Пресс, – 2009. – С. 136.
23. Румшиский Б.Л. Аппарат для описания морфологии флективных языков. – 2007. С. 45.
24. Сегалович И. Как работают поисковые системы в Интернете. – М.: Идея-Пресс, – 2009. – С. 239.
25. Сегалович И.В, Зеленков Ю.Г., Нагорнов Д.О. Методи порівняльного аналізу сучасних пошукових систем. – Суздаль, – 2009. С. 98.
26. Цейтин Г.С. Методы синтаксического анализа, использующие предпочтение языковых конструкций: модели и эксперименты //

Международный семинар По машинному переводу. М.: ВЦП. – 2005. – С. 131-133.

27. Цинман Л.Л., Сизов В.Г. Система Этап: процедуры ослабления синтаксических правил и их использование. Труды Международного семинара Диалог’2008 по компьютерной лингвистике и ее приложениям. – М.: Идея-Пресс, – 2008. С. 325.
28. Arasu A. Searching the Web. // ACM Trans. on Internet Technology, – № 1(1), 2009, p. 34-36.
29. Boguslavskij I.M., Tsinman L.L. A linguistic processor for natural language dialogue wi«th databases // Artificial Intelligence and Information-Control System of Robots 2009. / Ed. I. Plander. Amsterdam: Elsevier Science Publishers B.V. 2009, p. 273-276.
30. Chomsky N. Syntactic structures, s'Gravenhage. Computational Linguistics 2007. Vol. 11, N. 1-3, 2007, p. 116.
31. Ilyinsky S., Kuzmin M., Melkov A., Segalovich I. An efficient method to detect duplicates of Web documents with the use of inverted index WWW2007, 2007, p. 34-39.
32. Gerald Salton, James Allan, and Amit Singhal. Automatic text decomposition and structuring. Information Processing & Management, 32(2): 2006. c.127-138.
33. Melcuk I. Lexical Functions: A Tool for the Discription of Lexical Relations in a Lexicon // Lexical Functions in Lexicography and Natural Language Processing. Ed. By L. Wanner. Amsterdam (Philadelphia). 2006. P. 37 – 102.
34. Mizzaro S. How Many Relevances in Information Retrieval? // Interacting With Computers. – 2008, – № 10(3).
35. Yuwono B., Lee D. Search and Ranking Algorithms for Locating Resources on the World Wide Web / B. Yuwono, D. Lee // Proc. of the 12th International Conference on the Data Engineering. – New Orleans (Louisiana), 2006.