УДК 004.8

Глибовець А. М., Шабінський А. С., Ольшевський Р. Я.

ПОБУДОВА ПОШУКОВОГО РОБОТА УКРАЇНОМОВНИХ НАУКОВИХ МАТЕРІАЛІВ

У статті зроблено огляд та детальний аналіз основних проблем інформаційного пошуку. Представлена модель та програмна реалізація пошукового робота для текстів наукового та науково-публіцистичного стилю українською мовою.

Ключові слова: *інформаційний пошук, пошуковий робот, науковий стиль, науково-публіцистичний стиль.*

В статье сделан обзор и детальный анализ основных проблем информационного поиска. Представленная модель и программная реализация поискового робота для текстов научного и научно публицистического стилю украинским языком.

Ключевые слова: *информационный поиск, поисковый робот, научный стиль, научно-публицистический стиль.*

The review and detailed analysis of the main problems of information retrieval are covered in the article. The model and program implementation of the searching robot for Ukrainian texts written in scientific and journalistic style are represented.

Key words: *informative search, searching robot, scientific style, scientific-publicism style.*

Вступ

З моменту появи інформаційного пошуку (ІП) як напряму науки головною його проблемою незмінно є якісне задоволення інформаційної потреби користувача. Із появою концепції Семантичного Вебу він отримав нові перспективи і почав розвиватися у бік систем із імітацією розуміння змісту інформації. Англійська мова як міжнародна була і є головною мовою обміну інформацією світової спільноти. Засоби інформаційного пошуку, тим паче семантичного, в україномовному інформаційному просторі знаходяться у зародковому стані. Наразі постає проблема створення інформаційної пошукової системи (ІПС), яка б виконувала ефективний семантичний пошук україномовної наукової інформації.

Метою даної роботи є розробка моделі та програмної реалізації пошукового робота для текстів наукового та науково-публіцистичного стилю українською мовою. Важливим моментом досягнення цієї мети було дослідження та аналіз основних проблем створення означених систем, аналіз деяких існуючих підходів до вирішення даних проблем, застосування результатів аналізу до вироблення зазначеної моделі.

Запропонована модель та архітектура пошукового робота визначає підхід до імплементації кінцевого продукту за рахунок поєднання оптимальних існуючих рішень із розробкою необхідних компонентів, адаптованих під українську мову та кінцеве цільове використання. Кінцева імплементація запропонованої моделі буде забезпечувати принципово новий спосіб роботи користувача із україномовною науковою інформацією.

Основні проблеми інформаційного пошуку

Становлення інформаційного пошуку як самостійної науки починається у 1945 році із виходом новаторської статті Венівара Буша «Як ми можемо мислити» [1]. Поняття «інформаційний пошук» вперше використав у докторській дисертації Кельвін Мур у 1948 році, а з 1950 року термін почав інтенсивно вживатися у наукових розробках [2]. Наступні значні кроки було зроблено у 1960-тих роках, зокрема завдяки розробці системи SMART та методології оцінювання пошукових систем [3].

Користувач зацікавлений не просто віднайти множину документів, що відповідають його запиту (в ідеалі – інформаційній потребі), а й мати змогу відрізнити документи за ступенем їх відповідності запиту. Для цього у пошукових системах обраховуються точні показники, а результуюча множина документів упорядковується відповідно до обраної методики, що у ІП зветься *ранжуванням*.

Основною проблемою, що цікавить як користувачів, так і дослідників, є здатність пошукової системи знаходити релевантні документи. Проблема оцінки ефективності системи ІП ставиться у вигляді бінарної класифікації усіх документів колекції – на релевантні та нерелевантні. Релевантність оцінюється на основі інформаційної потреби користувача, а не на основі відповідності між запитом та знайденим документом. Ще однією важливою задачею ІП є забезпечення достатньої виразності мови запитів.

Попри всі досягнення у сфері обчислювальної техніки, задачі ІП вимагають створення ефективних систем із урахуванням і фактору складності обчислень.

Із моменту початку розвитку галузі ІП коло задач постійно розширювалось, і сьогодні, окрім згаданих центральних проблем, включає в себе ще декілька актуальних задач. Перш за все слід згадати проблеми класифікації та кластеризації документів. Не зважаючи на схожість кінцевої мети в обох задачах, їх суть зовсім різна. Класифікація як проблема штучного інтелекту полягає у віднесенні об'єкта до одного із заздалегідь визначених класів. Класифікація документів у текстовому ІП звужується на область текстової інформації і розглядає проблему віднесення документів до фіксованих класів на основі аналізу вмісту та машинного навчання. Застосовно до ІП, проблема кластеризації полягає у групуванні документів у підмножину кластерів за їх відповідністю певному запиту (запитам), або навіть деякій інформаційній потребі. Кластеризація є прикладом безнаглядового (англ. *unsupervised*) машинного навчання, на відміну від класифікації, що є прикладом наглядового (англ. *supervised*) навчання. У наглядовому навчанні бере участь людина-експерт, яка заздалегідь визначає розбиття на класи, а задача машини – відтворити процес класифікації за зразком експерта. У безнаглядовому навчанні експертний взірець відсутній. Класифікація та кластеризація документів є методами текстового аналізу, які слугують спільній справі – згрупувати документи за тими чи іншими ознаками схожості, аби документи в межах однієї групи були якомога більш подібними, а документи із різних груп чітко розрізнялися.

Решту задач ІП на сьогодні складають проблеми реферування документів, проектування та розробка мов запитів, прикладні питання проектування архітектур пошукових систем та користувацьких інтерфейсів, загальні питання моделювання.

Передбачається, що ми матимемо справу із україномовними текстами, переважно наукового та науково-публіцистичного стилю. Джерелами інформації може бути як «всесвітня павутина», так і множина довірених бібліотек тощо. У функціонуванні системи братимуть участь експерти, які будуть забезпечувати ті інтелектуальні функції, які система буде не в змозі забезпечити сама: аналіз, класифікація специфічної інформації, уточнення, покращення результатів пошуку, внесення нової інформації, ручне редагування бази знань тощо.

Взірцем ефективної ІПС нам бачиться така гіпотетична пошукова система, де процес визначення відповідності документа інформаційній потребі (а разом із цим і ранжування) виконується людиною, а решта необхідних процесів ІП – збір, індексація, класифікація, будь-яка механічна обробка інформації тощо – покладається на відповідне програмне забезпечення. Авжеж така система є лише гіпотетичною, але вона якнайкраще передає суть ідеалізованої пошукової системи, у якій інформаційна потреба користувача буде задоволена у повній мірі, звісно, за умови існування потрібної інформації.

Текстовий аналіз

Текстовий аналіз (ТА) означає видобування потенційно корисних і до цього невідомих знань із колекцій текстів. [4; 5] Із моменту появи ТА був частковим випадком аналізу даних (data mining). Аналіз даних передбачає роботу із структурованою інформацією, в той час як ТА – неструктурованою. У деяких підходах ТА як метод аналізу даних на колекціях неструктурованих текстів використовується для видобування структурованих наборів даних для подальшої обробки методами класичного аналізу даних. На противагу таким підходам існують методи аналізу знань у текстах. Такі методи базуються на тих фактах, що текст не є неструктурованим у загальному випадку. Навпаки, він побудований за складними, хоча й неявними, правилами – синтаксичними, семантичними, морфологічними тощо [4].

Текстовий аналіз приносить користь у всіх галузях, де потрібна ефективна робота із великими обсягами текстів. ТА у ІП може покращити точність та повноту пошуку, зменшити обчислювальну складність шляхом категоризації текстів відповідно до напрямків пошуку тощо. При створенні ІПС методи ТА необхідно використовувати аби автоматизувати процес створення бази знань, про що піде мова далі.

Задача класифікації є класичною задачею штучного інтелекту. Тим не менш, нагадаємо її у адаптації до текстів. Отже, є множина документів D та множина класів (категорій) C, і дано деяку навчальну вибірку документів, співставлених із класами, $\langle d, c\rangle \in D \times C$. Задача класифікації полягає у використанні деякого навчального алгоритму для знаходження функції класифікації, що співставляє документи класам: $\psi: D \to C$.

У [2] розглядається декілька навчальних методів з вчителем: за моделлю Байеса, за моделлю Бернуллі, класифікація Роккіо, метод найближчого сусіда, та деякі суміжні проблеми. Найбільшу увагу приділено методу опорних векторів, як одному із сучасних методів машинного навчання, який добре зарекомендував себе у задачах класифікації текстів.

Цікаво, що у системах ІП місце для машинного навчання є не лише у засобах класифікації документів. Машинне навчання добре підходить для визначення релевантності документів та ранжування результатів. У ІПС класифікацію можна застосовувати на етапі побудови онтологій для накопичення первинної бази документів в якості джерела для бази знань, можна згодом застосовувати для підтримання актуальності бази знань, коли нові документи з певної теми потрібно враховувати мірою надходження, аби зберігати повноту пошуку якомога вищою.

Прихований семантичний аналіз

Прихований семантичний аналіз (Latent Semantic Analysis) – математичний метод моделювання значення слів і фрагментів тексту шляхом аналізу репрезентативних текстових корпусів [6]. LSA є загальним методом аналізу текстів природною мовою. У контексті ІП LSA називають прихованою семантичною індексацією (Latent Semantic Indexing).

У LSA текстовий корпус зводиться до вигляду прямокутної матриці із рядками-термінами та стовпчиками-документами. Кожний елемент матриці містить значення, обраховане із частоти появи терміну у документі. Після особливої декомпозиції (сингулярний розклад, англ. Singular Value Decomposition) кожний документ представляється як сума векторів термінів у ньому. У такому поданні легко обчислювати відповідність між термінами, термінами та документами та між документами. Зазначимо, у ІП LSI не визнається винятковим методом індексації чи ранжування. Натомість, застосування LSI є дуже перспективним підходом до кластеризації у колекціях документів [2]. Маючи згадане подання термінів та документів як векторів, можна реалізувати кластеризацію термінів у концептуальному просторі (відповідність термінів) та кластеризацію документів (відповідність документів).

Великий репрезентативний корпус текстів розбивається на фрагменти, на розсуд розробника, це можуть бути абзаци, цілі документи тощо. Колекція зводиться до матриці фрагментів-термінів: стовпчики відповідають обраним фрагментам тексту, рядки – термінам у цих фрагментах тексту. Кожний елемент матриці містить значення частоти вживання терміна у фрагменті. Ми відразу обираємо документи у якості фрагментів тексту. Досить часто матрицю документів-термінів дещо змінюють, обраховуючи значення елементів як вагу терміна у документі (сублінійно, на зразок $\log(frequency + 1)$) та колекції в цілому (наприклад, зворотню частоту документа) [6].

Матрицю документів-термінів розкладають зі зниженням рангу, використовуючи сингулярну декомпозицію. При цьому серед усіх сингулярних чисел обирають k найбільших, решта покладається в 0. Результат є апроксимацією розмірності k оригінальної матриці. Кожний документ тепер поданий у вигляді вектора розмірності k. У більшості випадків k обирається значно меншим від кількості термінів у оригінальній матриці. Для більшості симуляцій мови $50 < k < 1000$ є оптимальним із найкращими значеннями в межах [250, 350]. Проте не існує методики чи теорії визначення точних оптимальних значень для k для кожного окремого застосування. Даний крок дає важливе виведення: різні значення для відповідності між кожними двома термінами залежно від того, чи вживалися вони в одному контексті чи ні [6].

Суть LSA полягає у тому, що оригінальна матриця документів-термінів після низькорангової апроксимації методом сингулярної декомпозиції, по-перше, має значно менший ранг, що спрощує подальші обрахунки, а по-друге, утворює концептуальний простір, де кожний документ і термін має числову характеристику відповідності тому чи іншому поняттю. Декомпозиція оригінальної матриці дає у результаті три нові матриці, дві з яких є матрицями документів та термінів відповідно. Кожний рядок (стовпчик) матриці термінів (документів) має розмірність k та містить значення відповідності терміна (документа) кожному із k понять. Кількість понять k може бути не більшою за ранг оригінальної матриці, а як правило обирається меншою. При зменшенні k повнота зростає, а при деяких k порядку сотень досягаються непогані показники для точності, тобто LSA вирішує проблему синонімів [2].

Коли семантичний простір у LSA побудовано, здебільшого не виникає потреби його переобраховувати аби додати нові документи чи терміни. Для нових термінів, наприклад, вектор можна обраховувати як середнє векторів принаймні 10 документів, у яких термін зустрічається. Для нових документів вектори обраховуються як сума векторів термінів у документі. Ці правила разом із розвитком обчислювальних потужностей усувають проблему, яка поставала у зв'язку із практичністю застосування LSA через припущення про необхідність частого перерахунку. Зокрема, ще у 2008 році корпус на 500 млн. слів обчислювався протягом 1 дня на кластері середньої потужності [6].

Зменшення розмірності оригінальної матриці до найважливіших вимірів фактично дає деяку можливість виокремити концептуальні знання з загального масиву неважливої інформації, і представити ці знання у зручному для обчислень концептуальному просторі. Така абстракція від непотрібних деталей і концентрація на головних поняттях вважається у LSA дечим подібним до нормального людського підходу та сприйняття світу [4].

LSA непогано вирішує проблему синонімів у текстах, але лише частково впорається із проблемою полісемії. Полісемія – явище багатозначності слів. Оскільки кожний термін позначається лише однією точкою у семантичному просторі, то різні значення слова будуть поєднані в деяке середньозважене, а тому будуть втрачені при пошуку. Така проблема вимагає раннього відстеження полісемії та додаткової категоризації слів із кількома значеннями, із множинним включенням до семантичного простору та урахування усіх значень слова. Щоправда, цей процес є мало дослідженим і потребує пошуку ефективних рішень [7].

LSA як метод текстового аналізу, на нашу думку, може відігравати важливу роль у аналізі колекцій документів для видобування знань і подальшої побудови онтологій. Важливо, що LSA є математично обгрунтованим формальним підходом.

Мовні аспекти ІПС

Розроблювана нами ІПС орієнтована на українську мову. Важливість імплементації моделі ІПС для української мови випливає із актуальної потреби у ефективному семантичному пошуку україномовної інформації. Ми стилістично обмежуємо область роботи ІПС тільки науковою та науково-публіцистичною інформацією. По-перше, пошук наукової інформації майже не розвинений в Україні, хоча у ньому є гостра потреба. По-друге, обробка лише наукової інформації дає змогу у подальших дослідженнях та удосконалення ІПС зосередитися на таких проблемах як автоматизована побудова онтологій та покращення

якості пошуку, а відкинути проблеми обробки неформальної лексики та структури речень, жаргону, сленгу тощо.

Складність роботи із українською мовою зумовлена тим, що серед слов'янських мов, не кажучи про англійську чи хоча б німецьку, українська мова майже не має підтримки з боку комп'ютерної лінгвістики. Так, української версії WordNet досі не створено. Корпус української мови далекий від своїх аналогів для російської та чеської мов, і фактично не доступний до кінцевого користувача. Ті спроби, які робляться у сфері комп'ютерної лінгвістики української мови, або не набувають достатнього розвитку, або не достатньо публічні, а відтак недоступні для консолідації зусиль. Ці фактори спричиняють гостроту проблем, пов'язаних із створенням ефективних пошукових систем україномовної інформації [8].

При імплементації моделі ІПС для української мови необхідно враховувати відмінюваність слів.

Пошуковий робот наукових матеріалів

Одними з базових складових частин інтелектуальних пошукових систем (*ІПС*) є програми збору та обробки інформації, які ще називають пошуковими роботами. Пошуковий робот («веб-павук», краулер, спайдер) – програма, що є складовою частиною пошукової системи і призначена для обходу сторінок інтернету з метою занесення інформації про них (ключові слова) у базу [9]. Такі програми ще називають «програми-павуки». З часом з'явилася спеціалізація пошукових роботів: *роботи-дятли* (що лише перевіряють «веб-сторінки» на доступність, «простукують» їх), *роботи-індексатори зображень*, *роботи-шпигуни*, *роботи-оглядачі* тощо.

Ми розглядатимемо роботів, першочерговим завданням яких є визначення рівня науковості матеріалів веб-сторінки та віднесення їх до певної галузі науки .

Пошуковий робот наукових матеріалів можна вважати фільтром, тобто перед внесенням інформації в базу даних пошукової системи, вона перевіряється на науковість, класифікується на відповідність категорії науки.

– ***Web crawling***

Нехай існує початкова база веб-посилань (URL) на ресурси (наприклад створена експертом). По кожному посиланню робот працює за такими правилами:

- Робот відвідує сторінку за посиланням і копіює її вміст подібно до веб-браузера.
- В скопійованому *html*-коді знаходить внутрішні та зовнішні посилання, шляхом пошуку входжень «*http://*», «*href=*» тощо і вносить їх в базу даних.
- Виділяє серед *html*-коду інформаційну частину тексту, шляхом видалення усіх тегів та іншого нерелевантного «сміття для подальшого аналізу.

– ***Визначення науковості тексту***

Існує кілька критеріїв для визначення науковості тексту [10]. Вони ґрунтуються на таких основних ознаках наукового стилю:

- Логічна послідовність викладу (використання складнопідрядних речень, причинно-наслідковий зв'язок, зв'язні слова тощо).
- Об'єктивність викладу (використання слова «ми» у значенні «я», чіткий поділ тексту на складові частини).
- Використання дієслів із загальним значенням оцінки, дієслів зі значенням становлення тощо.

Зрозуміло, що найпростіше для цього використати перевірку наявності в тексті універсального десяткового класифікатора. Але нам бачиться цікавим аналіз самого тексту на науковість. Класифікатор можна додати і до не наукового тексту.

Для програмної реалізації цього підходу необхідна участь експерта. Він формулює словник ключових слів, які характеризують вищезазначені ознаки науковості тексту. Цей словник представляється у вигляді XML-файлу, наприклад:

```
<word>перейдемо до</word>
<word>далі розглянемо</word>
<word>зупинимося на</word>
<word>повернемося до</word>
<verb>
    <infinitive>оцінювати як</infinitive>
    <secondperson>оцінюєш як</secondperson>
</verb>...
<word>- це</word>
```

За допомогою простого підрахунку кількості входжень цих ключових слів у тексті можна оцінити науковість тексту.

– ***Визначення предметної області***

Для тексту визначається частотність використання термінів поточної галузі знань згідно з XML-словником, що в результаті і є показником рівня науковості за даним науковим напрямом.

– ***Коефіцієнт науковості тексту***

Маючи показники науковості та категорії, ми можемо визначити належність тексту до наукових матеріалів. В результаті тестових випробувань нашого робота на ***1000*** наукових статей, ми дійшли висновку, що не раціонально відносити текст більш ніж до ***3*** наукових напрямів.

Статистичні дані аналізу на науковість (було оброблено 3111 статей, як наукових, так і ненаукових), приведено у таблицях 1 і 2:

Таблиця 1

Статистичні дані аналізу на науковість

Загальний рівень науковості	К-сть статей	Серед них наукових	%
< 0,6 %	1047	72	6,88 %
0,6 %-1,2 %	712	69	9,69 %
1,2 %-2 %	1197	942	78,7 %
> 2 %	155	140	90,32 %

Таблиця 2

Статистичні дані аналізу приналежності матеріалів до категорії «Економіка»

Рівень науковості за кат. «Економіка»	Кількість статей	Серед них у цій категорії	%
< 5 %	2509	14	0,56 %
5 %-10 %	118	21	17,8 %
10 %-15 %	317	243	76,66 %
> 15 %	167	135	80,84 %

За цими результатами можна зробити висновок, що порогом ознаки науковості може бути відсоток входження ключових слів S_g до всіх слів тексту більший або рівний 1,2 %.

Схожа ситуація з рівнями науковості за категоріями. Порогом S_i для них можна вважати 10 %.

Для покращення критерію відбору визначимо коефіцієнт науковості S таким чином:

$$S = S_g + \frac{\sum_{i=1}^{k} s_i}{k},$$

де k – *кількість* категорій визначення науки.

– ***Відмінювання слів***

Одною з проблем цього підходу є те, що експерти, добираючи слова для термінологічних словників, не вказують усі їхні відмінкові форми. Це занадто трудомістко. У той же час, в текстах не завжди використовуються лише інфінітиви дієслів та іменники в називному відмінку. Тому робот повинен вміти розпізнавати у тексті не лише саме вказане слово, але й усі його відмінкові форми.

Вирішенням цієї задачі міг би стати пошук спільнокореневих слів, але похибка при такому пошуку була б занадто великою. Тому ми розробили систему відмінювання слів: іменників за відмінками та числами, дієслів за часами та особами, та прикметників за особами та відмінками. Завдяки цій системі, експерту достатньо вказати початкову форму слова (деколи з деякими додатковими даними) замість усіх можливих його форм.

Для прикладу, щоб внести слово «прибуток» до XML-словника термінів для категорії «Економіка», адміністратору достатньо додати до нього такі рядки:

```
<noun>
    <word>прибуток</word>
    <gender>m</gender>
</noun>
```

В даному прикладі експерт використовуючи спеціалізований інтерфейс ввів слово – «прибуток», вказав, що це слово іменник <noun>, а також вказав чоловічий рід – <gender>m</gender>. Система зберегла надані відомості у вигляді XML коду.

Опишемо процес відмінювання. Спочатку визначається належність іменника до однієї з чотирьох відмін. Після цього визначається група: тверда або м'яка. Відповідно до відміни використовуємо одну з функцій відмінювання іменників, в нашому випадку «conjucateSecondDeclension». У її тілі визначаємо основу іменника, а далі, за правописом, до основи додаємо закінчення.

При відмінюванні прикметників спочатку визначається група прикметника, після цього виокремлюється основа. Далі, за правописом до основи додаємо закінчення відповідно до роду. Де потрібно (наприклад у родовому відмінку однини чоловічого роду «блідолиций»), основа міняється на «блідолиць» за допомогою відповідної функції.

Відмінювання дієслів реалізовано наступним чином. Для прикладу, розглянемо відмінювання дієслова «помилятися». Функція conjucateVerb визначить, що дієслово закінчується на «ся», а тому обріже основу і вважатиме інфінітивом «помиляти», а формою другої особи однини «помиляєш», відповідно збереже цю частку і додасть її до кожного слова результату (де потрібно замінивши на «ться»). Відповідно до дієвідміни, доконаності та групи реалізовано шість окремих функцій які відміняють дане дієслово за часами.

– ***Реалізація пошукового робота***

Пошуковий робот розроблений з використанням Java і MySQL в середовищі Eclipse. Він складається з трьох частин: експертний інтерфейс, пошуковий модуль і база даних. Опишемо їх детальніше.

Експертний інтерфейс використовується для внесення нових URL до бази даних та редагування критеріїв науковості та категорій знань.

В якості бази даних була обрана база MySQL, що вирізняється високою продуктивністю і безкоштовністю. В базі даних за кожним унікальним URL зберігається наступна інформація: унікальний номер, власне URL ресурсу; текст сторінки; булева змінна індикації статусу обробки; булева змінна належності науковому стилю; коефіцієнт науковості тексту; три змінні характеристики категорії знань і відповідно їм три змінні показники рівня науковості за цими категоріями.

Пошуковий модуль складається з парсера і аналізатора. Парсер бере з бази даних перший неопрацьований URL і завантажує його вміст. В тексті ресурсу виокремлюються нові URL і записуються в базу даних. Далі парсер очищує текст ресурсу від сміття і передає текст аналізатору. Аналізатор проводить перевірку на науковість та належність до категорій знань. Результати роботи аналізатора записуються в базу даних.

Схематично роботу пошукового робота зображено на рис. 1.

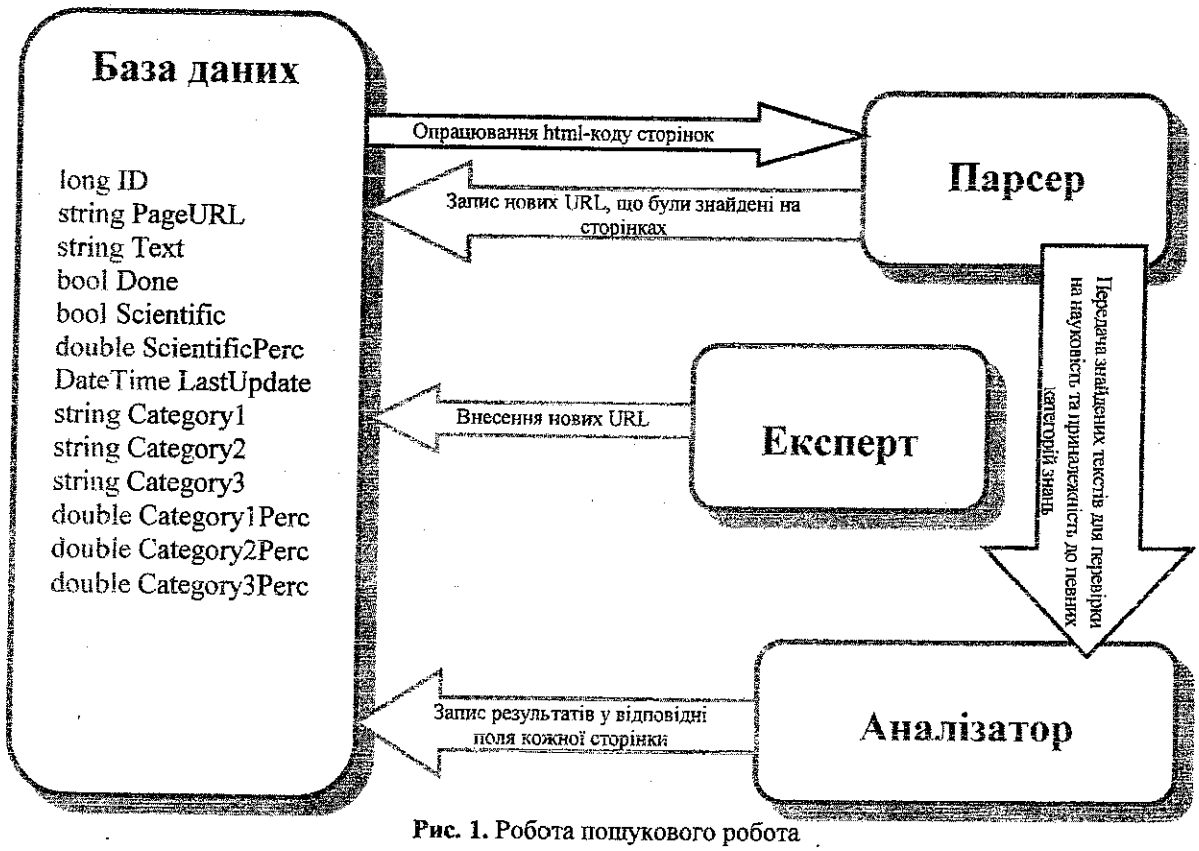

Рис. 1. Робота пошукового робота

Висновки

У даній роботі ми дослідили загальні проблеми створення інформаційних пошукових систем із наголосом на обробку україномовних документів. Було проаналізовано три основні проблеми у пошукових системах: ефективний збір та класифікація інформації, адекватна інтерпретація запитів користувача та оптимальний пошук україномовної інформації.

Методи текстового аналізу були виділені як ключові у автоматизації видобування знань у ІПС. З-поміж інших ми виокремили прихований семантичний аналіз, який у контексті ІП називають прихованою семантичною індексацією. Також обґрунтовано застосування методів класифікації для автоматичного визначення предметних областей документів.

За результатами аналізу була запропонована архітектурна модель та прогамна реалізація пошукового робота україномовної інформації наукового типу. У якості парадигми програмування для імплементації моделі рекомендовано об'єктно-орієнтований підхід, що забезпечить модульність та розширюваність системи.

Пошукові роботи наукових матеріалів – важливі складові частини ІПС. В нашій моделі їхнім першочерговим завданням є визначення рівня науковості документу та його належність до певної області науки. Показник науковості обчислюється шляхом визначення частоти (кількості) входжень зарезервованих ключових слів у текстах.

Ключові слова подаються для використання роботом у вигляді XML-словників. Розроблена система відмінювання слів: іменників за відмінками та числами, дієслів за часами та особами, та прикметників за особами та відмінками.

Практична апробація програмної реалізації показала високу точність визначення науковості документа та віднесення його до певної категорії науки за умови достатнього наповнення відповідних словників експертом.

Подальші дослідження та розробки доцільно спрямувати у напрямку створення україномовного WordNet, корпусів української мови та окремих підмножин, автоматизації створення онтологій для пошукових систем. Імплементація пошукових систем на основі запропонованої моделі матиме велике теоретичне та практичне значення для україномовного інформаційного пошуку.

ЛІТЕРАТУРА

1. Bush, Vannevar. As we may think. The Atlantic Monthly [Електронний ресурс]. – Режим доступу : http://www.theatlantic.com/magazine/archive/1969/12/as-we-may-think/3881/.
2. Manning, Christopher D., Raghavan, Prabhakar та Schütze, Hinrich. Introduction to Information Retrieval. New York : Cambridge University Press, 2008 [Електронний ресурс]. – Режим доступу : http://nlp.stanford.edu/IR-book/pdf/irbookonlinereading.pdf.
3. Singhal, Amit. Modern Information Retrieval: A Brief Overview. Bulletin of the IEEE Computer Society Technical Committee on Data Engineering. 2001 p. Text Knowledge Mining: An Alternative to Text Data Mining [Електронний ресурс]. – Режим доступу : http://singhal.info/ieee2001.pdf.
4. Sánchez, D., та ін. IEEE, 2008. 2008 IEEE International Conference on Data Mining Workshops [Електронний ресурс]. – Режим доступу : http://www.foibg.com/ibs_isc/ibs-10/ibs-10.pdf.
5. Stavrianou, Anna, Andritsos, Periklis та Nicoloyannis, Nicolas. Overview and semantic issues of text mining. ACM SIGMOD Record. – 2007. – Т. 36, 3.
6. Landauer, Thomas K та Dumais, Susan. Latent semantic analysis. Scholarpedia. 66072, 2008 p [Електронний ресурс]. – Режим доступу : http://www.scholarpedia.org/article/Latent_semantic_analysis.
7. Deerwester, S., та ін. Indexing by Latent Semantic Analysis. Journal of the American Society for Information Science. – 1990. – 41.
8. Анісімов А. В. Розробка методів автоматизованого розширення та добудови онтолігчних баз знань / А. В. Анісімов, М. М. Глибовець, П. П. Кулябко, О. О. Марченко, К. С. Лиман // Наукові записки НАУКМА. Комп'ютерні науки. – 2009. – Том 99. – С. 50–53.
9. Вікіпедія. Пошуковий робот [Електронний ресурс]. – Режим доступу : http://uk.wikipedia.org/wiki/Пошуковий_робот.
10. Гливобець А. М. Один підхід до побудови інтелектуальної пошукової системи / А. М. Гливобець, А. С. Шабінський // Наукові записки НАУКМА. Комп'ютерні науки. – 2010. – Том 112.– С. 26–30.

Рецензенти: Кондратенко Ю. П., д.т.н., професор;
Фісун М. Т., д.т.н., професор

Стаття надійшла до редколегії 20.12.2010 р.