УДК 004.89

ГЛИБОВЕЦЬ М.М.

СЕМАНТИЧНИЙ ПОШУК ІЗ ПОШИРЕННЯМ АКТИВАЦІЇ

У статті описано основні аспекти використання семантичного пошуку для підвищення ефективності Web та алгоритм семантичного пошуку із поширенням активації. Алгоритм має часову оцінку $O(|E| \cdot \log(|V|))$, а де E – потужність множини відношень графа екземпляра онтології а. V – кількість концепцій графу. Як показало тестування, успіх методу поширення активації є дуже чутливим до домену, але при правильно підібраній конфігурації, повнота пошуку зростає на 30-40 %.

The article clarifies the main aspects of semantic search usage for Web and semantic search with activation distribution enhancement. The algorithm has a time estimation of $O(|E| \cdot \log(|V|))$, where E is the potention of the relations set of the ontology entities graph, and V is the quantity of conceptions in the graph. The results of the tests showed that the activation distribution method is susceptible to domain, but when the configuration is properly chosen, the search effectiveness grows up to 30-40%.

Вступ

Ефективність Web безпосередньо залежить від технологій пошуку. На сьогодні превалює пошук за ключовими словами. Очевидні недоліки цього підходу – значна кількість нерелевантних результатів через потенційну полісемію фрази запиту, відсутність зв'язків між результатами пошуку, непрозорий працемісткий процес виведення нових знань із уже наявних.

Semantic Web (SW) – еволюційний крок у розвитку Web. Базовою технологією SW є онтології – категоріальні бази доменів; цеглинами SW є описи та екземпляри онтологій. Завдяки семантичному анотуванню веб-ресурси стають доступними для автоматичної обробки, саме тому SW можна порівняти із "глобальною базою даних". За рішенням W3C, наразі постала задача розробки стеку відкритих стандартів, серед яких уже чинні рекомендації мов опису онтологій RDF, RDFS, OWL, на черзі мова запитів до онтологій SPARQL та протокол обміну даними SW.

Новаторство SW полягає у використанні пошуку за концепціями: пошук сутності, яка відповідає фразі запиту в описах онтологій; пошук сутності в екземплярах онтологій, які анотують веб-ресурси і презентація віднайдених ресурсів користувачу. Семантичний пошук суттєво відрізняється від традиційного і може бути реалізованим через спеціальні пошукові паттерни, механізми виведення, індексування SW документів новими цікавими способами.

За [1], онтологія – формальна експліцитна специфікація концептуалізації. Онтологія впроваджує спільну категоріальну базу домену. Концептуалізація є абстрактною

спрощеною репрезентацією домену. Онтологія є специфікацією, бо формально визначає концептуалізацію.

Ресурси, анотовані категоріями із онтологій, є компонентами повторного використання. Завдяки онтологіям та логічному виведенню над ними, значення нових концепцій може бути виведеним із уже наявних. Тому розгортання SW тотожне побудові розподілених онтологій, реалізації рушіїв семантичного пошуку, засобів логічного виведення над онтологіями.

Онтології репрезентують спільну категоріальну базу домену. Рівень метаданих над Web (SW) утворюється екземплярами онтологій – множинами анотацій інформаційних ресурсів категоріями із визначень онтологій. Пошук за концепціями, вдосконалення інтеграції прикладних програм (онтології натомість XML-словників), впровадження інтелектуальних агентів – "прибуток" від онтологій. W3C розробила стек стандартів на підтримку онтологій – RDF, RDFS, OWL, SPARQL, це ознака того, що задля сумісності всі програмні продукти повинні підтримувати ці рекомендації. Онтології – ґрунт наступного рівня абстрагування, на який виходить комп'ютерна наука – сервісноорієнтованих архітектур. Онтології спростять публікацію, пошук та активування веб-сервісів, при такому підході програмне забезпечення буде мати будову динамічного конвеєра сервісів, а не статичного пакета компонент. Але, у першу чергу, онтології стануть на підтримку найпопулярнішого застосування Web – інформаційного пошуку.

Семантичний пошук

Розрізняють два різновиди пошуку: навігаційний та дослідницький.

У навігаційному користувач вводить до пошукової машини фразу чи комбінацію слів, яку потрібно знайти в документах. Запит у такому випадку не позначає жодної концепції, і пошукова машина виконує роль навігаційного інструмента у просторі документів, релевантних фразі запиту.

У дослідницькому користувач надає пошуковій машині фразу, яка напевне денотує концепцію чи відношення між концепціями. Власне, стосовно такого запиту потрібно зібрати інформацію. Результатом пошуку є набір ресурсів, які спільно висвітлюють значення концепції.

Технології пошуку за ключовими словами не вирішують завдання дослідницького пошуку. Завдяки семантичному анотуванню інформаційних ресурсів можна встановити концепцію, яка відповідає фразі запиту, в описах онтологій і визначити контекст, в якому вона потенційно може знаходитись. Наступний крок – пошук ресурсів, що відповідають цій концепції в екземплярах онтологій. Для звуження пошуку користувач заздалегідь може вказувати домен, задавати пошуковий паттерн (тобто шукати тільки у визначеному контексті). Такі інновації збільшать точність пошуку, введуть нові презентаційні можливості (наприклад у вигляді графа). Отже, семантичний пошук – це пошук за концепціями.

Співвідношення семантичного пошуку та пошуку за ключовими словами ілюструє рис. 1.

SW технологія – це набір засобів для розширення Web через надання інформаційним ресурсам формальної експліцитної семантики для кращої співпраці людей та машин [2]. SW буде містити ресурси, які відповідають не тільки медіа-об'єктам (веб-сторінки, малюнки, аудіо- та відеокліпи), але і персоналіям, місцям, організаціям, подіям. SW властиві декілька різних типів відношень між різними ресурсами (а не лише гіперлінки).

Визначимо аспекти SW технології, вагомі для семантичного пошуку. SW – це не пошук Web документів, а пошук Web відношень між ресурсами, які денотують об'єкти реального світу: персоналії, місця, події. Дані на веб-сторінці, приміром, є читабельними для людини; дані файлу у форматі RDF є машинно-інтерпретованими,

оскільки формально описують інформацію машинно-інтерпретованою мовою (RDF). SW містить багату машинно-інтерпретовану інформацію про ресурси.

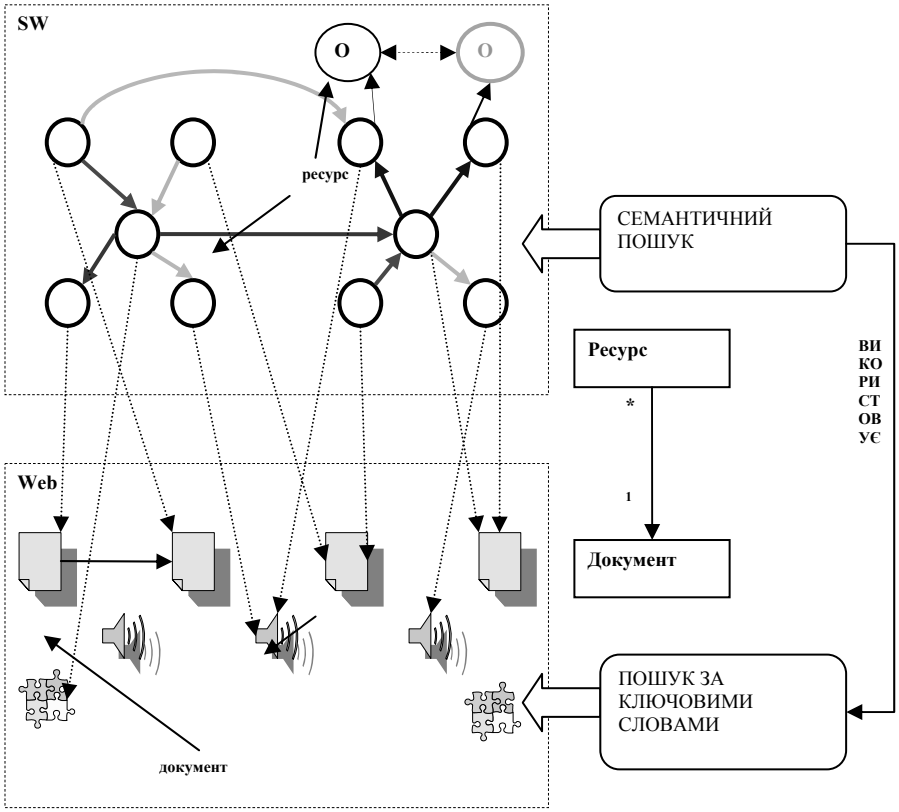

Рис. 1. Співвідношення семантичного пошуку та пошуку за ключовими словами

Інформацію щодо одного і того ж ресурсу можуть публікувати різні джерела, це провокує багато проблем. У результаті в світі, де хто завгодно може стверджувати що завгодно про кого завгодно, багато інформаційних джерел не викликають довіри. Тому інструменти семантичного пошуку повинні бути озброєні механізмами визначення об'єктивного рівня довіри до електронного джерела – найочевидніший спосіб – цифровий підпис електронних джерел.

Узагальнимо, які проблеми та за допомогою яких засобів вирішує семантичний пошук.

Проблема	Вирішення
Полісемія фрази запиту	Специфікація домену
Відсутність зв'язків між результатами пошуку	Лінкування ресурсів у RDF – графі
Брак засобів для формулювання контексту запиту, відношень між концепціями у запиті	Специфікація запиту через пошукові патерни

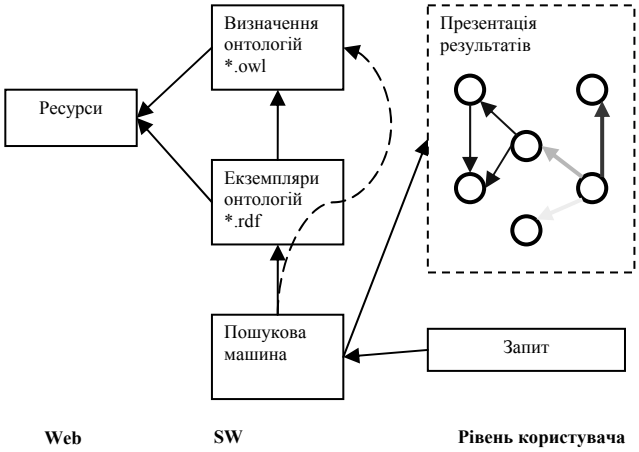

Рис. 2. Спрощена схема семантичного пошуку

Найбільш природним є семантичний пошук (рис. 2). анотованими ресурсами, але на сьогодні більшість документів не мають таких анотацій. Тому потрібно застосовувати просунуті методи текстової аналітики, розробити інструменти автоматичного семантичного анотування, анотувати документи видобутими концепціями. Найбільш відомим є інструментарій Semantic Analysis Workbench [3], IBM Webshpere Information Integration OmniFind Edition [4, 5] (це вирішення пропонує каркас Unstructured Information Management Architecture (UIMA).

Видобування інформації (Information Retrieval, IR) це сфера діяльності на перетині теорії інформації та комп'ютерної науки, котра стосується індексування та видобування інформації із гетерогенних інформаційних ресурсів. Термін IR був запропонований Муром у 1951 році; Мур пов'язував IR із "інтелектуальними аспектами" опису інформації та систем пошуку інформації [6].

IR-проблема – це проблема кластеризації, визначення того, які об'єкти, специфіковані користувачем, належать множині A, а які ні. По-перше, потрібно знати, які риси характеризують множину A, по-друге, які риси краще відрізняють ресурси у множині A від інших. Перший факт враховується частотою зустрічності терміна у документі (term frequency, tf), а другий враховується оберненою частотою зустрічності терміна в корпусі документів (inverse document frequency, idf).

У випадку семантичного пошуку видобування інформації повинне працювати на рівні семантичних анотацій ресурсів, щоб надавати адекватні результати у відповідь на концептуальні запити. Класична векторна модель цілком застосовна у цьому випадку і може бути адаптована через заміну компонентів, які входять до неї, як от: ключові слова, документи, запити. На концептуальному рівні немає таких синтаксичних відповідників, як релевантні терміни, хоча їхню роль можуть відігравати концепції із онтологій доменів. Тому семантичні документи можна репрезентувати як вектори, задані над гіперпростором усіх концепцій онтології, "вектори концепцій". Документи – це контейнери ресурсів, семантика яких описана через RDF-триплети, що позначають концептуальний зв'язок між ресурсами та вузлами онтологій.

Застосування моделі векторного простору дає переваги через узгодженість між запитом та способом презентації документа. Запити також повинні бути у формі векторів, заданих над гіперпростором концепцій онтологій.

Є два важливих аспекти, на які потрібно зважати при застосуванні векторної моделі на концептуальному рівні: формат запиту і припущення, на яких базується модель – ключові слова формують ортогональний базис для простору документів та запиту, а концепції онтологій не є взаємно незалежними, а навпаки, пов'язані різними типами

відношень, наприклад успадкуванням. Власне цей недолік можна ліквідувати через механізм навігації онтологій, який знаходить пов'язані концепції для запиту, наведений у підрозділі цієї роботи.

Алгоритм поширення активації

Оригінальним методом на підтримку семантичного пошуку є поширення активації – "spread activation". Поширення активації працює над екземплярами онтологій доменів. Відношенням між концепціями онтології – властивостям концепцій у термінах OWL присвоюються чисельні ваги. Поширення активації призначене для виявлення тісно пов'язаних концепцій у екземплярі онтології за множиною заданих концепцій та відповідних значень активації. Ці початкові значення є результатами традиційного пошуку, застосованого до ресурсів (вузлів RDF – графа), анотованих категоріями із онтології. Алгоритм активації поширення суттєво залежить від домену. Відносна вага шляху в графі до певного екземпляра концепції може бути оцінена тільки у добре визначеному контексті. Тому, залежно від домену, на роботу алгоритму активації поширення накладаються різні обмеження (максимальна довжина шляху, максимальна кількість вузлів, які розгортаються алгоритмом тощо) [7].

Новаторство такого підходу в інтеграції традиційного пошуку та технологій поширення активації.

Головна проблема традиційних пошукових машин у тому, що релевантність ресурсу встановлюється за ключовими словами. Щодо семантичного пошуку, то вимога бути ознайомленим із концепціями домену ускладнює висловлення користувачем його інформаційних потреб. Із цих причин доцільно поставити у відповідність кожній концепції домену множину ключових слів.

Через поширення активації поза результатами традиційного пошуку ми видобуваємо екземпляри концепцій, котрі пов'язані з концепціями, асоційованими із ключовими словами. Алгоритм поширення активації працює як "дослідник концепцій". При заданій множині активованих концепцій та спеціальних обмежень на поширення, активація протікає у мережі та виявляє концепції, тісно пов'язані із початковими. Кожному екземпляру відношення приписується чисельна вага, оскільки використання семантичної інформації поряд із символьною повинне покращити пошук. Таким чином, у роботі поширення активації працює із "гібридною мережею". Для автоматичного визначення ваги відношень пропонуються три метрики – кластерна, специфічності, комбінована.

Кластерна метрика. Ця метрика призначена на визначення подібності між екземпляром Cj та екземпляром концепції Ck.

$$W(C_j, C_k) = \frac{\sum_{i=1}^{n} n_{ijk}}{\sum_{i=1}^{n} n_{ij}}.$$

n_{ij} дорівнює одиниці, якщо екземпляри концепцій C_j та C_k пов'язані між собою та нулю у протилежному випадку. n_{ijk} дорівнює одиниці, якщо обидві концепції – C_j та C_k одночасно пов'язані із C_i, інакше – нулю. За цими міркуваннями, вага $W(C_j, C_k)$ означає відсоток концепцій, із якими пов'язана C_k, за умови, що C_j також знаходиться у відношенні із цими концепціями. Ця міра показує, що концепції, які знаходяться в багатьох спільних відношеннях із іншими концепціями, є більш схожими. Аналог показника $W(C_j, C_k)$ у IR моделі – df.

Метрика специфічності. Ця метрика відповідає показнику idf у IR. Метрика специфічності слугує для визначення специфічності відношення (наскільки часто воно є вживаним у домені) і задається такою формулою:

$$W(C_j, C_k) = \frac{1}{\sqrt{n_k}}.$$

Значення n_k еквівалентне числу екземплярів цього відношення, для яких концепція C_k є вузлом призначення (у термінах RDF-триплетів {<підмет>, <предикат>, <додаток>} – додатком). Тому вага відношення між концепціями C_j та C_k обернено пропорційна числу відношень, до котрих входить C_k. Чим більшою є кількість відношень, у яких C_k виступає додатком, тим менша вага кожного з цих відношень.

Комбінована метрика. Комбінована метрика є добутком кластерної та метрики специфічності (за зразком добутку df · idf у IR моделі).

Алгоритм

Алгоритм стартує із початкової множини екземплярів концепцій, здобутих традиційним пошуком. Цим вершинам присвоюються активаційні ваги (зазвичай вага активаційного вузла встановлюється в одиницю, а звичайного вузла – в нуль), під час поширення активуються інші вузли. Усі початкові вершини заносяться до черги з пріоритетами у не-зростаючому порядку значень ваг. Надалі із черги видобувається і обробляється вершина з найбільшим пріоритетом. Якщо поточний вузол задовольняє всі обмеження, то він поширює активацію на своїх сусідів. Нехай початкова вершина – i, а вершина призначення – j, тоді поширення активації до сусідів відбувається відповідно до такої формули, де I – вхід, O – вихід:

$$I_j(t+1) += O_i(t) \cdot w_{ij} \cdot f_{ij} \cdot (1-\alpha).$$

Функція ваги вершини i ($O_i(t)$) додається до поточної ваги вхідної вершини j, помножена на вагу відношення між поточною і наступною вершиною w_{ij} (кластерну метрику), відносну вагу відношення f_{ij} (метрика специфічності) та α – відсоток вже активованих відношень на шляху від початкової вершини до j. Значення всіх цих трьох множників залежать від домену. Поступово всі активовані алгоритмом вузли додаються до черги пріоритетів. Процес продовжується до тих пір, доки не досягнуто бажаного стану (наприклад, до знаходження заздалегідь вказаної кількості екземплярів концепцій або у черзі пріоритетів немає вершин для обробки). По завершенні процесу всі знайдені вершини упорядковуються за порядком активації. Зауважимо, що алгоритм обробляє кожну вершину лише один раз; у термінах часової складності становить $O(|E|)$, де E – потужність множини відношень графа екземпляра онтології. Операції над чергою пріоритетів мають часову складність $O(\log|V|)$, де V – кількість концепцій графа. Тому тотальна часова складність алгоритму поширення активації становить $O(|E| \cdot \log(|V|))$. Псевдокод алгоритму наведено на рис. 3.

Активаційні ваги. Зауважимо, традиційна пошукова машина рангує результати за рівнем релевантності запиту. Кожній вершині RDF-графа присвоюється дійсне число з інтервалу (0;1] – ранг. Це значення виступає в ролі активаційної ваги вузла для алгоритму поширення активації. Саме ця множина вузлів RDF-графа є активаційною множною для алгоритму поширення активації.

Обмеження. Однією із проблем алгоритмів поширення активації є те, що коли процес не контрольований, то як результат пошуку може бути відібраний неадекватний граф. На вирішення цієї проблеми скеровані обмеження.

Приклади обмежень:

- Обмеження на тип концепції: активуватись повинні тільки вершини певного типу.
- Обмеження розгортання: повинні відбиратись тільки ті концепції, котрі знаходять у певних відношеннях із більшою/меншою кількістю концепцій, ніж задано.
- Обмеження на довжину шляху: алгоритм поширення активації не повинен видобувати концепції, котрі знаходять на більшій відстані від вхідних концепцій, аніж задано (експериментально з'ясовано, що оптимальна довжина шляху знаходиться у межах від одиниці до трьох).

```
List spreadActivation(VertexPriorityQueue input)
  {
  List output; RestrictionsSet preRestrictions, postRestrictions;
  ActivationFuction activationFunc; double distanceDecayFactor; bool stop;
    while((input.isNotEmpty()) && (stop != STOP_SPREAD_ACTIVATION))
      {
      currVertex = input.removeMax();
      if(checkRestrictions(preRestrictions, currVertex) == true)
        {
          activation = activFunc.getActivation(currVertex);
          currVertex.visited = true;
          for(every edge e/Orig(e) == currVertex)
            {
              destVertex = e.getDestinaiton();
              edgeType = e.getType();
               deltaInput = activation * e.getWeight() *
edgeType.getWeight();
              deltaInput *= (1 – distanceDecayFactor);
              destVertexInput += deltaInput;
              destVertexActivation = activFunc.getActivation(destVertex);
              if(destVertex.visited == false)
                input.addVertex(destVertex);
            }
          outputQueue.insertVertex(currVertex);
        }
                stop = checkRestrictions(postRestrictions);
    }
  return outputQueue;
}
```

Рис. 3. Псевдокод алгоритму поширення активації

Схема семантичного пошуку із поширенням активації. Користувач задає пошукову фразу ключовими словами. припустимо, що усі документи мають семантичні анотації, із кожною концепцією асоційовано список ключових слів. Традиційна пошукова машина знаходить документи, семантичні анотації котрих містять екземпляри концепцій, заданих користувачем. Надалі над множиною здобутих семантичних анотацій (графа екземплярів концепцій) залучається до дії компонента поширення активації, і система презентує користувачу остаточні результати (рис.4).

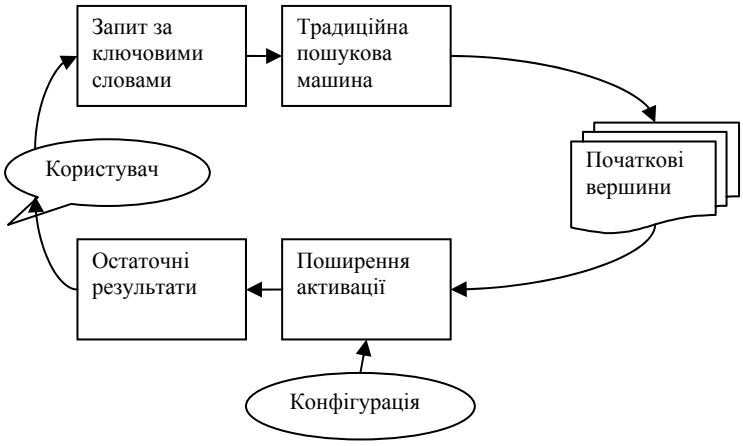

Рис. 4. Архітектура семантичного пошуку із поширенням активації

Висновок

У статті описано основні аспекти використання семантичного пошуку для підвищення ефективності Web та алгоритм семантичного пошуку із поширенням активації. Алгоритм має часову оцінку $O(|E| \cdot \log(|V|))$, де E – потужність множини відношень графа екземпляра онтології а. V – кількість концепцій графа.

Як показало тестування, успіх методу поширення активації є дуже чутливим до домену, але при правильно підібраній конфігурації повнота пошуку зростає на 30-40%.

ЛІТЕРАТУРА

1. Gruber T. It Is What It Does: The Pragmatics of Ontology. – 2003. – [Cited. 2006, January 12] – Available from <http://tomgruber.org/writing/cidoc-ontology.htm>.
2. Berners-Lee T. Web Architecture from 50,000 feet. – 1999. – [Cited. 2005, 2 December] – vailable from <http://www.w3.org/DesignIssues /Architecture.html >.
3. Jena – a Semantic Web Framework for Java. – Cited. 2006, 10 February] – vailable from <http://jena.sourceforge.net>.
4. IBM United States Software Announcement 205-002 (January 11, 2005): IBM WebSphere Information Integrator OmniFind Edition V8.2. – Cited. 2006, March 5] – vailable from <http://www.IBM.com/common/ssi /rep_ca/2/897/ENUS205-002/ENUS205-002.PDF>.
5. Hampp T., Lang A. Semantic search in WebSphere Information Integrator OmniFind Edition: The case for semantic search. (August 5, 2005). – Cited. 2006, March 20] – Aailable from <http://www-28.IBM.com/developerworks/db2/library/techarticle/dm-0508lang/>.
6. What the Semantic Web can represent? – [Cited. 2005, October 28] – Available from <http://www.semantic-conference.com/Semantic Technology Primer.htm>.
7. Rocha C., Schwabe D., Poggi de Aragão M. A Hybrid Approach for Searching in the Semantic Web. – 2004. – [Cited. 2005, October 8] – Available from <http://www2004.org/proceedings/docs/1p374 .pdf>.