ІНФОРМАЦІЙНІ СИСТЕМИ І ТЕХНОЛОГІЇ

УДК 004.853+004.832

МЕТОД АВТОМАТИЗОВАНОГО РЕФЕРУВАННЯ ТЕКСТОВИХ ДОКУМЕНТІВ З ВИКОРИСТАННЯМ ОНТОЛОГІЙ

Литвин В.В., Гайдін В.А., Пшеничний О.Ю.
Національний університет „Львівська політехніка”; vasyll@ukr.net

Постановка проблеми в загальному вигляді

Реферування тексту є однією найважливіших галузей для сучасних інформаційних технологій, оскільки кількість інформації, з якою працює людина, невпинно зростає й настає час, коли опрацювати весь необхідний матеріал просто неможливо. Використання комп’ютерів та створення мережі Internet дало змогу швидко отримувати та публікувати будь-яку інформацію. З одного боку, це значно прискорило пошук потрібних даних, збільшило ефективність роботи людини з документами та інформацією інших типів. Але водночас такий розвиток інформаційних технологій зумовив перехід суспільства до нового типу – інформаційного. У таких умовах обсяги інформації, з якими повинна працювати людина, зросли в десятки разів і знову перевищують людські можливості зі сприйняття та опрацювання цих даних. Тому робота в напрямі збільшення ефективності інформації є на сьогодні дуже важливою й актуальною. Але, незважаючи на це, досі не створено програмних продуктів, які б задовільно працювали в широкій сфері текстових документів та давали користувачу змогу значно зекономити свій час. Отже, проведення досліджень у напрямі автоматизованого реферування тексту є перспективним і необхідним для сучасного суспільства [1].

Аналіз останніх досліджень

Час опрацювання людиною текстової інформації залежить переважно від обсягів самого тексту, тому найбільш ефективним методом зменшення часу опрацювання текстових даних людиною є скорочення обсягів тексту з якнайменшими втратами його змісту. Цей процес називається реферуванням.

Існуючі на сьогодні системи автореферування широко використовують математичні методи опрацювання тексту, значно відстаючи лінгвістичною складовою алгоритмів, що не дає змоги досягти високої якості роботи прикладних систем. Такі інструменти, як функція Autosummarize в Microsoft Office, системи IBM Intelligent Text Miner, Oracle Context і Inxight Summarizer (компонент пошукового механізму Altavista), безумовно, корисні, але їх можливості обмежені виділенням і вибором оригінальних фрагментів з початкового документа і з'єднанням їх у короткий текст. Підготовка ж короткого викладу передбачає опис основного змісту тексту, і не обов'язково тими самими словами.

Основною метою цієї роботи є проектування інформаційної системи автоматизованого реферування тексту, яка б змогла добре працювати з текстами різного обсягу, жанру та складності термінології.

Аналіз процесу автоматизованого реферування текстів

Процес автоматизованого реферування тексту – це створення коротшої версії вихідного тексту за допомогою комп’ютера. Результат цього процесу – текст, що містить основний зміст вихідного. Системи автоматизованого реферування текстів широко використовуються під час текстового пошуку.

Існує дві основні парадигми реферування – витягнення інформації та її перефразування. Перший метод полягає у відкиданні всієї неважливої інформації з вихідного документа. Таким чином, на виході залишається "сухий залишок" документа, його суть. Недоліком такого методу є те, що принцип "із пісні слів не викинеш" діє не лише для поетичних творів – інколи декілька абзаців документа можуть формулювати одну дуже просту думку, але таким чином, що викинути жодне речення не виходить – втрачається змістовна інформація. У такому випадку на допомогу приходить друга парадигма, яка передбачає перефразування вихідного тексту, формування того самого змісту, але іншими словами. З точки зору реалізації, перший метод простіший, однак з точки зору ефективності роботи він гірший за другий. Саме другу парадигму ми й будемо розглядати в цій роботі.

Створення реферату тексту повинно відштовхуватися від того, для кого цей реферат призначений. Існує два основних види рефератів – призначений для читання людиною та призначений для опрацювання комп'ютером. Перший вимагає правильної побудови речень, використання художніх зворотів, другий – чіткого викладення ідей вихідного тексту. На сьогодні не існує систем автоматизованого реферування, які б могли створювати справді якісні реферати першого типу, однак на практиці вони потрібні набагато менше, ніж другий тип. Тому в цій роботі переважно буде розглядатися другий тип реферування.

Процес автоматизованого реферування тексту можна умовно поді на лити такі етапи (рис. 1):

1) початковий аналіз тексту, його розбиття на розділи, абзаци, речення;
2) семантичний розбір тексту, приведення його у форму, зручну для обробки комп'ютером;
3) аналіз змісту документа, при якому визначаються основна тема документа, ключові слова, відкидаються надлишкова та непотрібна інформація;
4) утворення реферату: з інформації, отриманої на попередньому етапі, утворюється реферат вихідного документа згідно із заданими обмеженнями.

Рис. 1. Етапи автоматизованого реферування тексту

Розглянемо вищезгадані етапи детальніше. Перший етап, початковий аналіз тексту, передбачає виділення структури документа. Будь-який документ має деревоподібну структуру – вершиною є сам документ, який складається із розділів, що, у свою чергу, містять підрозділи. На найнижчому рівні такого дерева лежать речення, прості і складні, які можна розбити на прості. Речення природної мови являє собою закінчену думку. Абзац, утворений кількома реченнями тому, що вони несуть якусь спільну думку. Те саме стосується й розділів. Таким чином, граматична структура документа сильно допомагає

під час його аналізу, мало того, без неї він би був практично неможливим. Завдання розбиття тексту на структурні одиниці є відносно нескладним і вже не раз вирішувалося – для цього використовують різноманітні алгоритми пошуку за шаблонами, наприклад скінченні автомати, що працюють з регулярними виразами. Основною вимогою для тексту, що піддається реферуванню, якраз і є чітка структура. На цей момент ніхто не ставить завдання реферувати неграмотно написані тексти без чіткої структури, оскільки вони ні для кого не становлять значного інтересу.

Наступний етап – семантичний аналіз тексту – включає в себе процес перетворення одиниць тексту в певну форму, що зручна для опрацювання машиною. У реченні виділяються підмет, присудок та інші одиниці, за допомогою онтології визначаються поняття, які використовуються в реченні, після чого воно записується у вигляді певного виразу, який оперує об'єктами онтології та встановлює зв'язки між ними, які задаються проаналізованим текстом [2; 3]. Подібний аналіз може здійснюватися не лише на рівні окремих речень, а й на рівні більших одиниць тексту, коли речення має сенс лише у своєму контексті. Загалом семантичний аналіз можна виразити у вигляді певної функції, яка приймає на вхід текст і повертає його в заданій системою формі. На вищому рівні функція розбиває вхідний текст на складові та викликає себе ж від цих частин, у подальшому оперуючи не текстом, а результатом їх виконання.

Етап аналізу змісту тексту передбачає визначення основної тематики документа, виділення основних ідей, відкидання зайвої інформації. Цей етап – найскладніший, оскільки система повинна прийняти рішення про те, яка саме інформація важлива. Часто це зробити не так просто, тому може бути утворено декілька версій реферату, кожна з яких відповідає певній тематиці, що була описана в тексті. Однозначний реферат можна утворити лише у випадку чіткої спеціалізації тексту на певній темі, але й за умови існування вимог до теми реферату з боку користувача. На цьому етапі широко використовується онтологія. Важливо також враховувати вимоги до кінцевого реферату, які часто задають міру, за допомогою якої визначається, який обсяг інформації повинен бути залишений.

Етап формування кінцевого реферату включає в себе процес зворотного перетворення інформації з внутрішньої системної форми в текст або ж певний стандартний вигляд, передбачений інтерфейсом системи. Так, наприклад, вихідний реферат навіть простого неформатованого тексту може бути документом у форматі XML, який значно зручніший для подальшого опрацювання комп'ютером-користувачем.

Таким чином, два основних етапи реферування тексту включають у себе використання онтології. Проблема побудови цієї онтології та роботи з нею буде розглянута далі.

Алгоритм роботи проектованої системи

Розроблена система складається з декількох модулів, що опрацьовують текст і виконують певні підготовчі операції, модуля виділення інформації, що вважається рефератом, а також модуля перетворення цієї інформації з внутрішнього формату системи в текст для виведення користувачеві.

Першим етапом роботи системи є розбивання тексту на частини (абзаци та речення). Це допомагає далі працювати з логічними частинами тексту. Для проведення реферування потрібно визначити основні терміни тексту. Визначення предметної області, до якої відноситься наданий текст, значно спрощує це завдання, але часто користувачу важко віднести текст до певної галузі, і він може зробити неправильний вибір. До того ж часто інформація, подана в тексті, стосується декількох предметних областей одночасно, що значно ускладнює вибір. Отже, необхідно визначити слова, що вживаються в тексті найчастіше, за винятком загальновживаних слів. Проблему створюють численні словоформи й відмінки, наявні майже в усіх мовах, у тому числі в українській. Тобто

просте порівняння слів на рівність, виходячи із сучасних реалізацій мов програмування, не дасть бажаного результату.

Ця проблема вирішується використанням словників словоформ. У цій роботі використовується словник Hunspell. Він дає змогу дуже швидко утворити потрібну форму заданого слова, якщо воно міститься в словнику. Для знаходження первинних форм слів було розроблено програму, яка на основі словника Hunspell згенерувала всі варіанти вживання слів і сформувала новий словник – словник первинних форм слів. Цей словник значно більший за розмірами, але дозволяє швидко виконувати операцію пошуку первинної форми слова й економити "дефіцитні" обчислювальні ресурси.

Знаючи кількість вживання кожного поняття (незалежно від відмінку, часу, роду та інших. характеристик слова), можна визначити, про що цей текст і, відповідно, які слова є найважливішими в ньому. Просте цитування найкращих за певним критерієм речень, яке зараз широко використовується в аналогічних системах, сформує незв'язний і багато в чому незрозумілий читачу реферат. Йому, скоріше за все, доведеться звертатись до оригіналу для отримання повнішої інформації про зміст окремих речень. Такий варіант не надто прискорить його роботу. Головною проблемою, що стоїть на шляху створення гарного реферату, є нездатність комп'ютера оперувати поняттями – він може працювати зі словами як з послідовностями символів, а не як з ідентифікаторами сутностей.

У цій роботі пропонується залучити до реферування тексту механізм семантичних мереж та онтологій. Процес побудови онтології описаний у наступному підрозділі. Кожне речення містить у собі закінчений вислів. Провівши синтаксичний та семантичний розбір речення з використанням онтологій предметних областей, можна визначити його зв'язки між словами та словосполученнями, виявити надлишкову інформацію, а також зв'язки з іншими реченнями, проаналізувавши займенники та слова-посилання.

Речення, в якому проведений семантичний розбір, легко подати у вигляді простої семантичної мережі, яка повністю описуватиме його структуру й за якою легко здійснювати пошук зв'язків з іншими реченнями. Вказана семантична мережа матиме зірчасту структуру, в основі якої лежатиме присудок та підмет речення, а додаткові члени речення (додатки та обставини) являтимуть собою гілки графа. Під час побудови семантичних мереж (СМ) усіх речень абзацу можна виконати поєднання самих мереж новими зв'язками, утворивши семантичну мережу абзацу. Аналогічно, об'єднанням СМ абзаців утворюється СМ тексту загалом.

Після побудови такого графа можна вільно отримати з нього будь-яку інформацію, що була описана в тексті. Для цього тільки треба знайти відповідні вершини СМ і їх залежні вершини та проаналізувати структуру зв'язків між ними. Якщо ж потрібно цю інформацію подати у вигляді тексту, то потрібно перевести інформацію, представлену в СМ, у текстовий вигляд. Для цього потрібно сформувати речення на основі інформаційних зв'язків понять, збережених у СМ. Щоб будувати коректні за структурою речення, необхідно мати опис будови речень потрібної мови, тобто коректних послідовностей частин мови, що формують цілісне речення. Маючи вершину семантичної мережі – поняття, яке необхідно описати, можна виділити пов'язані з ним характеристики, модифікатори та залежні поняття, погруповані за вживанням у реченнях первинного тексту, і побудувати кілька речень, що відповідають опису цього поняття в тексті. Таке завдання сьогодні не викликає значних труднощів реалізації при заданій структурі генерованих речень. Єдиною складністю є відмінювання слів та перетворення в потрібну словоформу. Це завдання вирішується використанням вищезгаданого словника Hunspell.

Останнім завданням, яке необхідно вирішити перед формуванням кінцевої форми реферату, є вибір послідовності опису матеріалу в рефераті. Оскільки він генерується повністю автономно від початкового тексту, то провести з ним прямий зв'язок, за аналогією до статистичного опрацювання, не вдасться. Тому пропонується запам'ятати розподіли частоти вживання термінів у тексті для кожного терміну. Набір понять, що

вживається найчастіше в певному фрагменті тексту, відображає загальну думку цього розділу тексту. Послідовність викладу інформації в рефераті повинна відповідати первинному тексту. Отже, під час генерування реферату потрібно послідовно для кожного розділу тексту описати його основний зміст. Якщо поняття, що фігурують там, пов'язані з іншими поняттями цього розділу, то опис продовжується рекурсивно. У випадку, якщо зустрічається поняття, яке відноситься до подальших розділів, можна зробити на них посилання й не описувати цей термін відразу. Якщо стек неописаних понять став порожнім, потрібно перевірити, чи всі важливі поняття поточного розділу внесені в реферат і, якщо так, переходити до наступного фрагменту тексту. При цьому поріг важливості понять визначається коефіцієнтом реферування, який вказав користувач.

Якщо згенерований таким чином реферат значно відрізняється за обсягом від потрібного користувачу, можна модифікувати поріг важливості понять і провести генерацію реферату знову. Ця процедура виконується доти, доки бінарним пошуком не буде знайдено величину, яка відповідає необхідному для користувача обсягу реферату, який і буде повернуто йому в якості результату роботи програми.

Аналіз процесу побудови онтології для автоматизованого реферування

Онтологія зберігається в пам'яті комп'ютера й у будь-який момент може бути доповнена новими об'єктами або зв'язками. Для правильного та ефективного функціонування онтології необхідне виконання декількох умов:

1) реалізація програмної бібліотеки, яка б могла представляти онтологію в пам'яті, зчитувати опис онтології з файла та зберігати її у файл;

2) бібліотека повинна використовувати ефективні алгоритми, швидкодія яких дає змогу працювати з онтологіями великих розмірів;

3) файли, у яких зберігається опис онтології, повинні бути у форматі, що використовується й іншими програмними засобами для роботи з онтологіями: для забезпечення можливості роботи із сторонніми онтологіями [4].

Ефективним методом вважається встановлення зв'язків із сторонніми онтологіями за допомогою мережі – підхід, що пропагується парадигмою Semantic Web.

Отже, модуль роботи з онтологією повинен складатися з таких підсистем:

- підсистема перетворення онтології із загальноприйнятого формату у внутрішній та навпаки;
- об'єктна модель для представлення сутностей онтології в пам'яті комп'ютера;
- підсистема пошуку об'єктів та зв'язків;
- підсистема додавання об'єктів та зв'язків;
- підсистема контролю за цілісністю онтології;
- механізм об'єднання онтологій з декількох вихідних файлів та зовнішніх онтологій з мережі;
- інтерфейс для роботи з користувачем, у даному випадку системою автоматизованого реферування.

Оскільки формати, у яких зберігаються та передаються описи онтологій, мають відповідати певним стандартам, то для роботи з ними їх потрібно опрацювати й подати у форматі, передбаченому архітектурою системи. Забігаючи наперед скажемо, що буде використовуватися стандарт OWL [5], як один із найпоширеніших і найефективніших, а отже, ця підсистема буде являти собою дещо вдосконалений механізм для роботи з XML.

Об'єктна модель для подання онтології в пам'яті – це бібліотека класів об'єктно-орієнтовною мовою програмування, яка дає змогу повною мірою відобразити всі елементи онтології в пам'яті комп'ютера.

Підсистема пошуку об'єктів та зв'язків – це компонент, що забезпечує навігацію по онтології у відносно швидкому режимі.

Підсистема додавання об’єктів та зв’язків – в онтологію в будь-який момент можна додати об’єкт чи зв'язок. Видалення елементів стандартом не передбачене.

Підсистема контролю цілісності виявляє колізії, що з’являються в онтології, коли додаються нові елементи. Згідно з принципами, що декларуються стандартом OWL, суперечності, що утворюються при цьому, мають опрацьовуватися системою-користувачем онтології згідно з підходом, який передбачений принципами її функціонування.

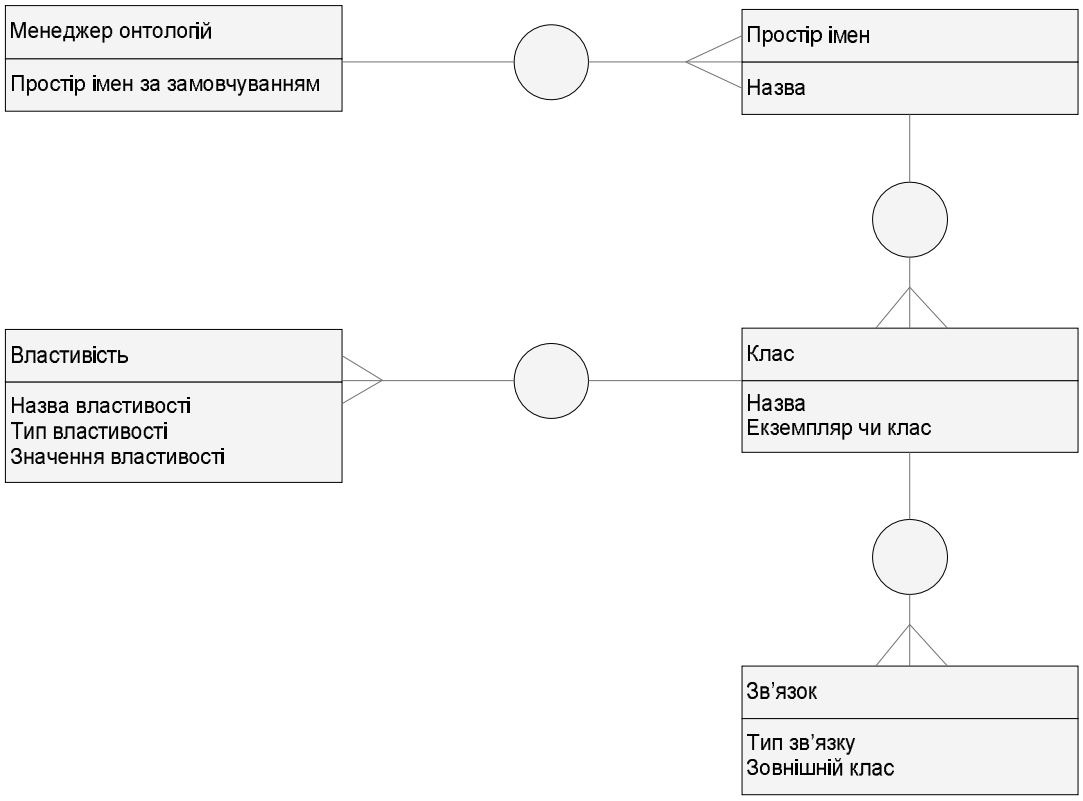

Рис. 2. Діаграма сутність-зв’язок, що ілюструє архітектуру розроблюваної бібліотеки класів

Механізм об’єднання декількох онтологій в одну – це потужна можливість, що дає змогу зливати багато онтологій у одну велику, тим самим забезпечуючи її повноту. Завдяки Semantic Web цей процес став як ніколи простим та ефективним – зовнішню онтологію можна просто завантажити з мережі Інтернет і бути впевненим, що вона буде подана в тому самому форматі, в якому представлена онтологія, якою оперує система.

Інтерфейс для роботи з користувачем – це відкрита бібліотека класів та методів для роботи з ними, яка дає змогу системі-користувачу оперувати онтологією так, як це їй потрібно.

Таким чином, після аналізу всього вищесказаного можна дійти висновку, що гарним рішенням для архітектури бібліотеки буде архітектура, зображена на рис. 2.

На цьому рисунку зображена архітектура, що складається з п’яти сутностей:

1) менеджер онтологій – сутність, що оперує наявними онтологіями, об’єднуючи їх в одну велику; менеджер містить у собі множину сутностей типу «простір імен»;

2) простір імен – це сутність, яка об’єднує в собі множину об’єктів онтологій; необхідна для забезпечення стійкості від конфліктів імен у межах однієї онтології;

3) клас – основний об'єкт онтології; може бути як класом об'єктів, так і конкретним екземпляром, в останньому випадку не може мати класів і об'єктів-нащадків; містить множину властивостей та зв'язків з іншими класами;

4) властивість – функція, що задає відношення між даним класом і певним літералом, який не є об'єктом онтології;

5) зв'язок – функція, що задає відношення між даним класом і зовнішнім класом.

Дві останні сутності мають багато спільного, тому на етапі реалізації можуть бути об'єднані в єдину шляхом створення абстрактного об'єкта, від якого вони наслідують ряд властивостей.

Висновки

У ході виконання цієї роботи було спроектовано систему автоматизованого реферування тексту з використанням онтологій. Спроектована інтелектуальна система реферування тексту використовує статистичні методи аналізу слів тексту для вибору основних понять реферату, проводить семантичний розбір речень з метою виявлення надлишкової інформації в самих реченнях, а також несловесних зв'язків між реченнями. З отриманих даних будується семантична мережа тексту, яка й опрацьовується алгоритмом реферування. У випадку створення якісної системи автоматизованого реферування, що зможе працювати з текстами різних жанрів, типу та складності, вся сфера пошуку текстової інформації зазнає серйозних змін. Сам пошук потрібних даних стане менш рутинним і ним можна буде фактично знехтувати. Проблеми в роботі з інформацією будуть зводитись до її наявності та перетворення в людські знання.

Література

1. Хан У. Системы автоматического реферирования / У. Хан, И. Мани. // Открытые системы [Електронний ресурс]. – Режим доступу до журналу: http://www.osp.ru/os/2000/12/178370/.

2. Даревич Р.Р. Метод автоматичного визначення інформаційної ваги понять в онтології бази знань / Р.Р. Даревич, Д.Г. Досин, В.В. Литвин. // Відбір та обробка інформації. – 2005. – Вип. 22(98). – С. 105-111.

3. Даревич Р.Р. Оцінка подібності текстових документів на основі визначення інформаційної ваги елементів бази знань / Р.Р. Даревич, Д.Г. Досин, В.В. Литвин, З.Т. Назарчук // Исскуственный интеллект. – 2006. – № 3. – С. 500-509.

4. Литвин В.В. Інтелектуальні системи підтримки прийняття рішень на основі адаптивних онтологій / В.В. Литвин, Р.О. Голощук // VI міжнародна науково-практична конференція „Математичне та програмне забезпечення інтелектуальних систем". – Дніпропетровськ, 2008. – С. 208-209.

5. Даревич Р.Р. Застосування інформаційних технологій для координації наукових досліджень / Р.Р. Даревич, Д.Г. Досин, В.В. Литвин, Л.С. Мельничок. – Львів: СПОЛОМ, 2008. – 236 с.

Інформація

Ахкубеков А.А. Контактное плавление металлов и наноструктур на их основе. – М. : Физматлит, 2008. – 152 с.

В монографии обобщены результаты теоретических и экспериментальных исследований в области физики контактного плавления твердых растворов с металлами и электропереноса в контактных прослойках. Рассмотрен механизм начальной стадии контактного плавления на наноуровне. Описано влияние малых примесей щелочных металлов и постоянного электрического тока на скорость контактного плавления.