УДК 004.421

К.т.н., доцент Петрашенко А.В., магістрант Клюско В.В.

Національний технічний університет України
«Київський політехнічний інститут»

МОДЕЛЬ КОМПОЗИТНИХ АТРИБУТІВ ТЕКСТОВИХ ДОКУМЕНТІВ

Abstract

Andriy V. Petrashenko, assoc. prof., PhD; Vitaliy Klyusko, student

The composite attribute model for text documents

This paper concerns the task of optimal work with attribute of text documents. Particular work with attributes which consists of the same data but have different text representation. Proposed composite attribute model resolve same problem in work with such attribute. The composite attribute model is a way of represent attribute value and set of functions to work with attributes.

Вступ

Сьогоденне глобальне інформаційне середовище містить величезні об’єми неструктурованої документальної інформації. Постійне зростання кількості текстових даних та відсутність уніфікованих засобів формалізації змісту документа значно ускладнюють автоматизацію зберігання і пошук релевантної інформації. Зокрема, пошук за атрибутами, які по суті однакові, але можуть мати різне текстове представлення.

Постановка задачі

Мета дослідження полягає в розробці узагальненої моделі композитних атрибутів та розширенні застосування атрибутів в документах. Запропонована модель дозволяє більш гнучко представляти та працювати з атрибутами текстових документів.

Модель композитних атрибутів

Модель композитних атрибутів являє собою спосіб представлення атрибутів та набір операцій, виконуваних над цими атрибутами. Модель передбачає ефективну роботу з неструктрованими текстовими масивами даних.

Характеристикою неструктурованості текстової інформації будемо вважати відсутність у документів спільних ознак, таких, як, наприклад, дата видання документа, автор тощо, за якими документи можна було б формально розділити на певні групи. Тому, як джерело інформації для структурування можуть виступати лише атрибути документа. В такій інтерпретації документ – це множина значень всіх його атрибутів[1].

В основі такої моделі лежить поняття композитного атрибуту. Композитний атрибут – це атрибут, який складається з більш ніж одного компоненту типу даних. Прикладами таких атрибутів можуть бути повне ім’я, яке складається з прізвища, ім'я, по батькові особи та адреса, яке складається з назви міста, вулиці, поштового коду тощо.

Завдяки тому, що композитний атрибут містить множину значень, за його допомогою можна описувати дані, які мають одне й теж значення, тобто є тотожними атрибутами, але можуть по різному представлятись в тексті. Наприклад, атрибут “Автор” документу може записуватись в різних форматах: ”Чехов А.П.”, “Антон Павлович Чехов” та інших варіантах. Застосовуючи композитний атрибут, такі значення будуть рівними, що посилює зв’язки між текстовими документами.

Розроблена модель комплексних атрибутів передбачає ефективну реалізацію на основі використання моделі зберігання даних ”Entity-Attribute-Value”[2,3]. В такому випадку документ подається у вигляді пов'язаних між собою таблиць, в яких містяться атрибути та їх значення відповідно.

Операції над значеннями атрибутів

Нехай A — множина всіх можливих атрибутів документів D_i, тоді документ D_i можна подати як кортеж, що складається з множини атрибутів $A_j \in A$ документа та їх значень V_j:

$$D_i = <A_j : V_j>$$

Вибірки потрібного компоненту композитного атрибуту відбувається по ключу: $V_i = A_j$ *(“*Ключ*”)*.

Операції над рядковими типами:

Concat(str1, str2, …), Length(str), Substring(str, pos, len), Lowercase(str), Uppercase(str), Capitalize(str)

Операції над числовими типами:

арифметичні операції : + , - , *, /

Операції з датою та часом:

Weekday(date), DayName(date), MonthName(date), Peroid(date1,date2), Year(date), Month(date), Day(date), Hour(time), Minute(time), Second(time)

Реалізації базових режимів обробки документів з використанням моделі

Пошук

Задача атрибутивного пошуку зводиться до використання операції вибірки з текстового сховища W[1], всіх документів з вказаними значеннями атрибутів. Наприклад, якщо потрібно знайти документи, по значенню композитного атрибуту “Автор” - “Чехов А.”, результатом буде: *W.where(A(“*Прізвище*”)=“*Чехов*”) ∩ W.where(*Substring *(A(“*Ім’я*”),*0 ,1 *) = “*А*”) ∩ W.where(*Substring *(A(“*Побаткові*”),*0 ,1 *) = “*П*”)*. Рис.1 ілюструє даний пошук.

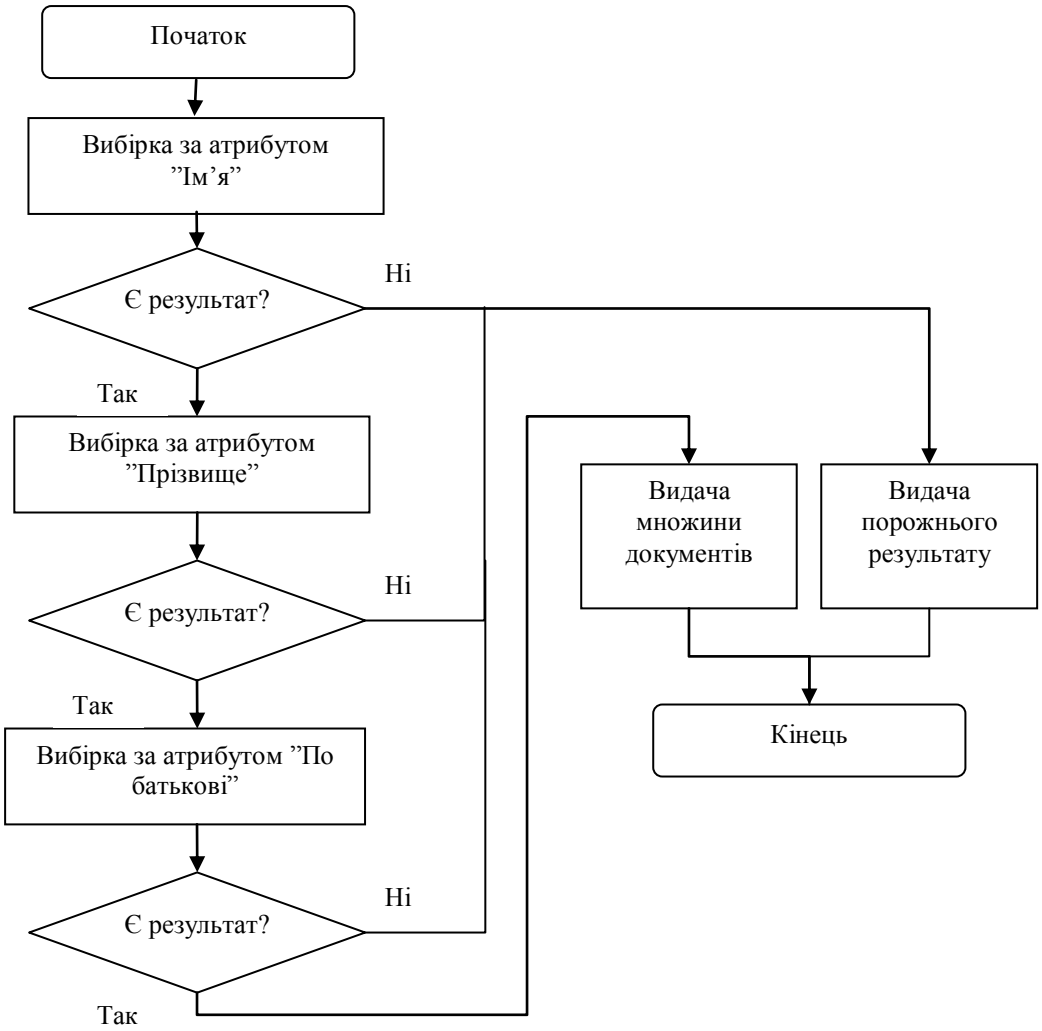

Рис.1. Алгоритм пошуку за комплексним атрибутом ”Автор”.

Каталогізація

Побудова каталогу документів у загальному випадку включає вирішення двох задач: побудову дерева каталогу та визначення набору документів, які відносяться до вибраної гілки. Побудова дерева зводиться

до визначення всіх наявних значень атрибуту на певному рівні ієрархії каталогу. Наприклад, для документів з атрибутом “Автор” - “Шевченко Т. Г.” так *“*Рік*”* = *“*2009*”* обчислення списку гілок за атрибутом *“*Назва*”)* виглядатиме так:
*(W.where(A(“*Прізвище*”)=“*Чехов*”)* ∩
*W.where(*Substring *(A(“*Ім’я*”),*0 ,1 *)* = *“*А*”)* ∩
*W.where(*Substring *(A(“*Побатькові*”),*0 ,1 *)* = *“*П*”)* ∩ *W.where(A(“*Рік*”)=*
*“*2009*”)).values(A(“*Місто*”))*

Друга задача — знаходження набору документів, які відносяться до обраної гілки, зводиться до фільтрації:
*W.where(A(“*Прізвище*”)=“*Чехов*”)* ∩
*W.where(*Substring *(A(“*Ім’я*”),*0 ,1 *)* = *“*А*”)* ∩
*W.where(*Substring *(A(“*По батькові*”),*0 ,1 *)* = *“*П*”)* ∩
*W.where(A(“*Рік”*)*= “2009”*)* ∩ *W.where(A(“*Місто*”)=* “Київ”*)*
Таким чином, показано можливість реалізації традиційних методів обробки текстових ресурсів.

Висновки

Дослідження та розробка нових способів роботи з атрибутами текстової інформації є достатньо актуальною науково-технічною задачею. У ході дослідження розроблено узагальнена модель комплексних атрибутів, додано комплексність до поняття атрибут. Застосування такого підходу дозволяє більш гнучко організовувати дані, підвищує повноту та релевантність пошуку в неструктурованих текстових сховищах.

Література

1. *Mykhailyuk A., Zamiatin D., Petrashenko A.* Unstructured Data Warehouse Processing System Based on an Uniform Set of Functions // Proceedings of the 4-th International Conference ACSN-2009 “Advanced Computer Systems and Networks: Design and Application” – Lviv. – 2009. - P. 117-119
2. *Jennings, Roger*, "Retire your Data Center", Visual Studio Magazine Feb 2009. – P. 14-25
3. *Dinu, Valentin; Nadkarni, Prakash*, "Guidelines for the effective use of entity-attribute-value modeling for biomedical databases", Int Journal of Medical Informatics 76 (Nov-December 2007). – P. 769–779