УДК 519.876.5

МЕТОДИ РОЗВ'ЯЗУВАННЯ ЗАДАЧІ НЕЧІТКОГО СПІВСТАВЛЕННЯ ЗАПИСІВ В РЕЛЯЦІЙНИХ БАЗАХ ДАНИХ

Порплиця Н.П.[1)], Франко Ю.Ю.[2)]
Західноукраїнський національний університет
[1)] к.т.н., доцент; [2)]магістрантка

І.Актуальність задачі

На сьогоднішній день при роботі з великими обсягами даних виникає проблема, яка полягає в тому, що вам потрібно співставляти два чи більше набори даних, які мають лише назви. А також у них відсутні загальні ідентифікатори або будь-яке інше поле, по якому можна співставити ці записи. Тому для розв'язування такої задачі, у загальному випадку потрібно застосовувати методи чисельної оптимізації [1]. Наприклад, клієнт отримує звіти від різних компаній, які далі потрібно об'єднати у один і зробити аналітику, тобто потрібно співставити ці записи.

Для розв'язування таких задач використовують «нечітке співставлення» - процес використання алгоритмів для визначення наближеної (отже, нечіткої) подібності між двома наборами даних. А також термін, що використовуються для опису процесу об'єднання двох наборів даних, які не мають спільного унікального ідентифікатора.

Ця задача є загальновідомою, досить складною і її важко вирішити систематичним способом, особливо, коли мова йде про великі набори даних. Звичайно, можна застосувати емпіричний підхід із використанням операторів Excel та vlookup, але це вимагає «людського втручання». Отож якість даних є предметом підвищеної уваги у бізнесі, медицині, освіті, а їх невідповідність у інформаційних системах є причиною значної втрати доходів.

ІІ.Огляд існуючих методів розв'язування задачі

Розглянемо відомі методи для розв'язування описаної в попередньому розділі статті задачі. Відстань редагування Левенштейна (ВРЛ) [1]: цей метод використовує редагування відстані для порівняння подібності двох рядків. Редагування відстані є звичайним показником текстової подібності, що визначає мінімальну кількість вставок, видалень та підстановок одного символу, необхідних для зміни одного рядка на інший (тобто зробити два рядки рівними) [2]. Відстань редагування несиметрична, вона містить 0<=d (x, y) <=max (| x |, | y |) [3] , де x показує кількість символів у двох рядках , d (x, y) - міра відстані. Наприклад: 'відстань' і 'відстнань'. Просте порівняння за характером свідчить про те, що всі літери після букви „д" неправильні. Однак останні три («ань») виглядають правильно, незважаючи на розбіжність. Мінімальна відстань до рядка - 3. Та насправді, часто існує кілька відповідей, оскільки для обчислення відстані редагування може існувати більше одного мінімального набору перетворень. Кожна трансформація називається «вирівнюванням» і представляє можливе пояснення допущеної помилки [4]. Для пошуку оптимальної відстані редагування використовується алгоритм динамічного програмування. Обчислювальна складність часу цього алгоритму значною мірою впливає на затрати часових ресурсів, особливо для великих баз даних.

Відстань редагування Левенштейна для двох рядків (s1, s2) можна відмітити як LED (s1, s2). Побудовано метрику подібності між двома рядками, яка варіюється від 0 до 1,0 за допомогою нормованої формули:

$$Sim(s1, s2) = \left(1 - \frac{LED(s1, s2)}{\max len(s1, s2)}\right) \quad (1)$$

,

де $\max len$ позначає максимальну кількість символів у цих двох рядках довжиною $s1$ та $s2$;

LED – це відстань редагування Левенштейна, яка є мінімальною кількістю видалень, вставок та підстановок, необхідних для перетворення спірного рядка в представлений.

З цієї формули значення подібності 1 означає, що два рядки абсолютно однакові, тоді як значення 0 вказує на дуже малу подібність.

Однак проблема алгоритму Левенштейна для визначення схожості двох рядків полягає в тому, що вона не може дослілити, яке з двох спірних імен краще узгоджується із представленим рядком, коли два генерують однаковий LED, але з різними алфавітними конфігураціями. Зіставлення кількісних показників, таких як «подібність» з лінгвістичними концепціями: «велика подібність», «невелика подібність» і «мала подібність», щоб їх можна було використовувати в правилах прийняття рішень для визначення ступеня збігу між двома записами, є процесом, що складається із 2 етапів: визначити кількісну міру відповідного атрибута порівняння та присвоїти число від 0 до 1, щоб показати, наскільки сильно міра порівняння атрибута пов'язана з мовною концепцією.

Наступним відомим методом для розв'язання задачі є приблизна відповідність стрічки (approximatestringmatching), яка часто називається нечітким співставленням - це техніка пошуку стрічок, які приблизно відповідають шаблону (а не точно). Проблема наближеного збігу рядків, як правило, ділиться на дві підзадачі: пошук приблизних збігів підстрічок всередині даної стрічки та пошук стрічок словників, які приблизно відповідають шаблону.

Точність відповідності вимірюється кількістю примітивних операцій, необхідних для перетворення стрічки в точну відповідність. Це число називається відстанню редагування між стрічкою і шаблоном. Звичайними примітивними операціями є[5]:

вставка: Sofserve→Softserve
видалення: Sofftserve → Softserve
заміна: Saftserve → Softserve

Деякі приблизні збіги також вважають транспозицією, при якій 2 літери у стрічці міняються місцями:

транспозиція: Sotfserve → SoftServe

Одним із можливих визначень наближеної проблеми зіставлення стрічок є наступне: дано стрічку з шаблоном P = p1p2... pm та текстову стрічку T=t1t2…tn, знайти підстрічку, яка з усіх підстрічок T має найменшу відстань редагування до шаблону P. Підхід повного перебору полягає у обчисленні відстані редагування до P для всіх підстрічок T, а потім вибору підстрічки з мінімальною відстанню. Однак цей алгоритм теж має високу обчислювальну складність $O(n^2 * m)$.

Висновок

Таким чином у роботі розглянуто проблему нечіткого співставлення записів у реляційних базах даних. Проведено огляд та аналіз відомих методів її розв'язування, а саме: алгоритмЛевенштейна та алгоритм нечіткого співставлення. У ході цього аналізу було виявлено недоліки цих методів для великих баз даних. Тому у подальших дослідженнях, увагу буде сконцентровано настворенні додатковихфункцій для забезпечення нечіткого співставлення атрибутів за типом даних.У цій статті описується власне рішення проблеми нечіткого співставленняатрибутів реляційної бази даних та його корисність для вирішення безлічі проблем бізнесу, що дозволяє автоматизувати завдання, які раніше вимагали«людського втручання» в процес аналізу даних.

Список використаних джерел

1. Levenshtein VI. Binary codes capable of correcting deletions, insertions andreversals. Soviet Physics Doklady 1966;10(8):707–10.
2. Gu L, Baxter R, Vickers D, Rainsford C. Record linkage: current practice and future directions. CSIRO mathematics and information science[Електронний ресурс] – Режим доступу до ресурсу: http://citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.14.8119
3. G. NavarroA guided tour to approximate string matching ACM Comput Surv, 33 (1) (2001), pp. 31-88
4. MacKenzie IS, Soukoreff RW. A character-lever error analysis technique forevaluating text entry methods. NordiCHI 2002(October):19–23.
5. Baeza-Yates, R.; Navarro, G. (June 1996). "A faster algorithm for approximate string matching". In Dan Hirchsberg; Gene Myers (eds.). Combinatorial Pattern Matching (CPM'96), LNCS 1075. Irvine, CA. pp. 1–23. [Електронний ресурс] – Режим доступу до ресурсу: https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.42.1593