УДК 681.5:519.8

С.М. Москвіна
О.М. Москвін
М.В. Плис

МОДЕЛЬ СЛАБОСТРУКТУРОВАНИХ ДАНИХ ВЕЛИКИХ РОЗМІРНОСТЕЙ

Вінницький національний технічний університет

Анотація

В роботі запропонована модель слабоструктурованих даних великих розмірів BDC на базі графових структур та алгоритми ефективного їх зберігання, пошуку та аналізу. Проведено дослідження переваг використання розробленої моделі на прикладі оцінки ефективності операції пошуку. Було показано, що дана модель завдяки агрегації та використанні графової структури забезпечує високу швидкість обробки даних, що дозволяє застосовувати її для обробки BDC.

Ключові слова: модель слабоструктурованих даних, дані великих розмірів, BDC, графові моделі .

Abstract

There was developed a model of semi-structured data based on graph and algorithms for storing, searching and data analytics. There was an experiment to study the effectiveness of developed method based on searching operations. The method using aggregations and graph structure allows to obtain huge speed of searching and allows to use it for handling BDC.

Keywords: model, graph, big data handling.

Вступ

В наслідок автоматизації та комп'ютеризації процесів різного роду діяльності людини, в світі суттєво зросли обсяги інформації, що призвело до того, що в останні десятиліття приріст даних відбувається в експоненціальній формі і загальний їх обсяг може наближаться до декількох десятків Терабайт. Прикладами таких даних можуть бути медіа дані, курси акцій, які фіксувалися протягом десятиліть, історичні статистичні і метрологічні показники, дії та дані користувачів в мережі Інтернет та інше.

При обробці даних великих розмірностей (Big data capacity (BDC)) виникають дві задачі, які потребують розв'язку. По-перше, це задача структуризації BDC, які зазвичай неструктуровані та зберігається в текстовому вигляді. По-друге, це задача обробки неструктурованих даних, тому що існуючі програмні системи не пристосовані до обробки великих обсягів даних, оскільки розраховані на невеликі об'єми із визначеною структурою, у звязку з чим виникає необхідність в розробці нових моделей BDC та алгоритмів для їх обробки та аналізу [1]. Метою даної роботи є розробка моделі слабоструктурованих даних великих розмірностей для ефективного їх зберігання та обробки.

Результати досліджень

В залежності від природи даних та їх структурованості для збереження, обробки та представлення даних обирається певна структурна модель даних. Структуровані дані зазвичай представляються реляційною або мережевою моделями даних. Найрозповсюдженішою є реляційна модель даних, яка полягає в збереженні даних у реляційній базі даних (БД), в яких дані представляються набором відношень, операції над якими визначаються реляційною алгеброю. Проте така модель даних не підходить для неструктурованих даних, а також має обмеження в продуктивності. Іншим структурованим представленням даних є мережева модель даних, в якій дані подаються як сукупність об'єктів різного рівня, де кожен об'єкт може бути зв'язаний з іншим, але недоліком такої структури є велика складність схеми БД, а також складність обробки даних для кінцевого користувача.

В даній роботі запропонована модель слабоструктурованих даних на базі графових структур. Так, якщо вхідні дані надходять у систему з джерел, кількість яких складає N, множина всіх джерел як $S = \{s_1, s_2, \dots, s_N\}$, то кожне джерело даних містить певну кількість записів, що можна представити наступним чином:

$$R_{S_k} = \{r_{k_1}, r_{k_2}, \dots, r_{k_m} \mid s_k \in S\}. \quad (1)$$

Записи складаються з характеристик (полів) $P = \{p_1,\ p_2, \dots, p_L\}$, де L – максимальна можлива їх кількість. Характеристики діляться на ключові та другорядні, наявність яких в записах не є обов'язковою. Кожний запис є унікальним у своєму джерелі. Унікальність запису гарантується комбінацією його обов'язкових характеристик. Так наприклад, множину характеристик, які унікально ідентифікують запис у джерелі S_a можна описати як:

$$P_d = \left\{ p_{d_1}, p_{d_2}, \dots, p_{d_g} \;\middle|\; p_d \in P \right\} \tag{2}$$

Всі можливі комбінації, які ідентифікують ідентичний запис, виражаються множиною $\{P_{d_1}, P_{d_2}, \dots, P_{d_i}\}$. Допустимо, щоб отримати значення характеристики P_a з джерела S_b потрібно виконати операцію $X(P_a, S_b)$. Тоді множину всіх даних у сховищі можна представити формулою (3):

$$O = \begin{bmatrix} X(p_1, s_1), X(p_2, s_1), \dots, X(p_l, s_1) \\ X(p_1, s_2), X(p_2, s_2), \dots, X(p_l, s_2) \\ \dots \\ X(p_1, s_n), X(p_2, s_n), \dots, X(p_l, s_n) \end{bmatrix} \tag{3}$$

За умовою задачі необхідно знайти всі джерела, в яких присутні задані характеристики (4):

$$P_{dx} = \{s_1,\ s_2, \dots, s_t \mid s \in S\} \tag{4}$$

На наш погляд таку структуру даних зручно зберігати в деревоподібній структурі, де на кожному рівні дерева знаходиться множина можливих значень певної характеристики вхідних даних, яка буде представлятися вузлами графа. В якості прикладу побудуємо граф, який показаний на рисунку 1, де вхідні дані описуються трьома характеристиками $\{P_1,\ P_2, P_3\}$ відповідно, тому граф складається з трьох рівнів. Кожна характеристика (рівень графа) описується множиною свої значень. Так, наприклад, характеристика P_1 може приймати значення із множини $\{V_{11},\ V_{12}, V_{13}, V_{14}\}$, P_2 з множини $\{V_{21},\ V_{22}\}$, а P_3 з множини $\{V_{31},\ V_{32}\}$.

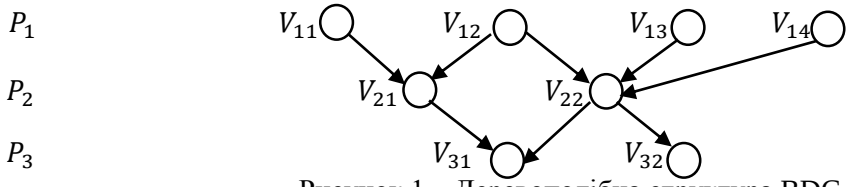

Рисунок 1 – Деревоподібна структура BDC

В цьому випадку операція завантаження об'єктів в графову структуру відбувається ітераційно. Процедура завантаження описується виразом:

$$O = O_e \cup O_n = \{o \mid o \in O_e \vee o \in O_n\} \tag{5}$$

Операція завантаження представляє собою операцію доповнення множини існуючих даних O_e множиною даних як завантажуються O_n. В результаті буде отримана множина об'єктів O_g, яка складається з існуючих даних (даних до завантаження) O_e та завантажених O_n. Так, наприклад, в процесі завантаження до існуючих даних, які представленні графом на рисунку 2(а) нових даних, які показані на рисунку 2(б) будуть отримані дані, які описуються графом з рисунку 2(в). Операція пошуку BDC являє собою віднаходження джерела, в якому знаходиться об'єкт або множина об'єктів із заданим критеріям (характеристикам) пошуку. Критерії пошуку задаються у вигляді множини характеристик $P_s = \{P_1,\ P_2, \dots, P_l\}$ кожний елемент якої, відповідає певному рівню.

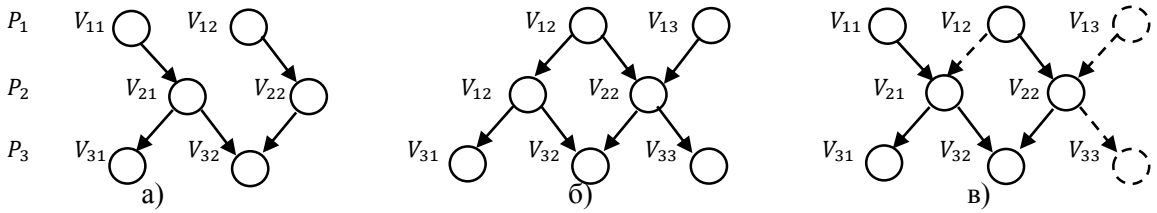

Рисунок 2 – Приклад операції завантаження BDC у вигляді деревоподібної структури

Операція пошуку можна представити операцією перетину значень характеристик існуючих даних та значень характеристик за якими проводиться пошук (6):

$$O_s = P \cap P_s = \{p \mid p \in P \wedge p \in P_s\} \quad (6)$$

В результаті буде отримано множину об'єктів O_s характеристики якого, зійшлися з пошуковими характеристиками P_s на кожному рівні. Припустимо, що є множина даних, що описуються графом з рисунку 3, і якщо пошук заданий множиною $\{V_{13}, V_{32}\}$, то будуть обрані об'єкти що належать графу, який виділений від основного графа штрих-пунктирними лініями.

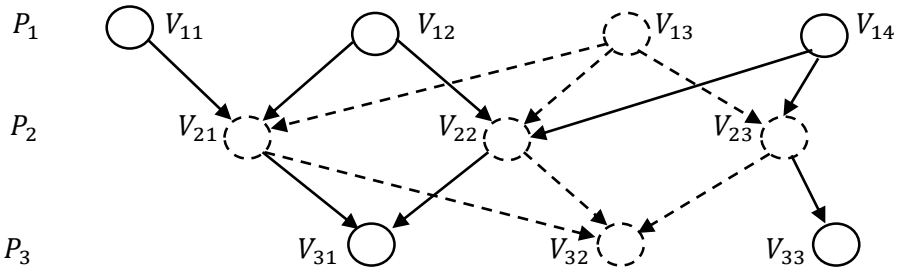

Рисунок 3 – Приклад операції пошуку BDC у вигляді деревоподібної структури

Операція передбачає покроковий рух по графу зверху вниз, на кожному рівні якого відсіюється *m-1* значень кожної характеристики. Складність пошуку в такому дереві залежить від глибини дерева і в не значній мірі від кількості вузлів на кожному рівні дерева.

Для оцінки ефективності запропонованої в роботі моделі BDC було проведено експериментальне дослідження на основі порівняння ефективності використання запропонованої моделі і реляційної моделі. Результати досліджень представлені в таблиці 1.

Таблиця 1 – Результати експерименту

Кількість завантаже-них файлів	Розмір файлів у файловій системі, Gb	Загальна кількість записів у файлах, млн.	Розмір бази даних, Gb		Час виконання запиту, мс	
			MySQL	Graph	MySQL	Graph
1	1,2	7,6	1,3	2	9	3
2	2,4	15,2	2,5	2,4	2	3
3	3,6	21	3,5	2,7	2	3
4	4,8	28,6	4,8	2,9	5	3
5	6	36,23	6,5	3,1	5	3
10	12	70	11,4	3,3	13	3
15	18	118	17	3,6	12	4
20	27	154	23	3,8	30	3

На рисунку 4 показані результати дослідження залежності часу виконання операцій пошуку BDC від кількості записів для реляційної бази даних та розробленого методу.

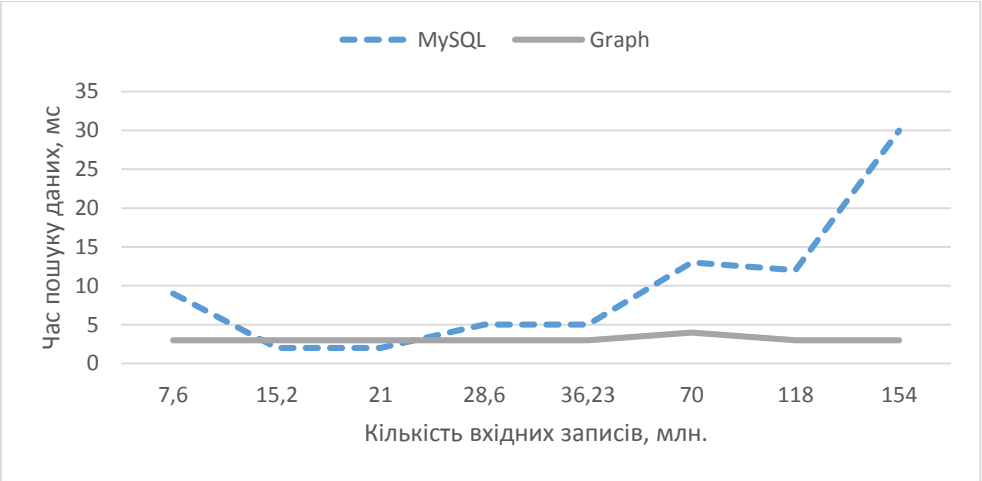

Рисунок 4 – Залежність часу виконання пошуку BDC від кількості записів

Результати експерименту вказують, що об'єм даних в реляційні моделі прямопропорційний об'єму даних у файловій системі. Збереження даних у запропонованної в роботі моделі є більш ефективним, оскільки зростання об'єму інформації у графовій структурі має логарифмічний характер, про що свідчать дані експерименту. Із результатів експерименту видно, що обсяг 20 файлів у файловій системі становить 27 Gb, а у запропонованій структурі – 3,8 Gb. Даний показник має велику перевагу для зберігання BDC, оскільки дозволяє більш оптимально використовувати дисковий простір. На рисунку 4 спостерігається прямолінійна залежність часу пошуку даних у графовій структурі від кількості записів, в той час як реляційна модель показує більш нестійку поведінку. З рисунку видно, що при кількості записів більше ніж 36 млн. час на виконання запиту у реляційній моделі швидко зростає, що не є прийнятним в обробці BDC. Тобто запропонована модель є ефективнішою за двома показниками.

Висновки

В роботі запропонована модель слабоструктурованих даних великих розмірів BDC на базі графових структур та алгоритми з використанням цієї моделі для ефективного їх зберігання, пошуку та аналізу. Проведено дослідження переваг використання розробленої моделі на прикладі оцінки ефективності операції пошуку, результати якого показали, що дана модель завдяки агрегації та використанні графової структури забезпечує високу швидкість обробки даних, що дозволяє застосовувати її для обробки BDC.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1.Москвін О., Плис М. Особливості розробки розподіленої системи обробки генетичної інформації // Тези доповідей XLV регіональної науково-технічної конференції професорсько-викладацького складу, співробітників та студентів ВНТУ. – 2016. – С. 3

Москвіна Світлана Михайлівна – к.т.н., професор, кафедри комп'ютерних систем управління, Вінницький національний технічний університет, Вінниця, e-mail: moskvina@ukr.net:

Москвін Олексій Михайлович — к.т.н., e-mail: moskvin.aleksey@gmail.com;

Плис Максим Валентинович — магістрант кафедри комп'ютерних систем управління, Вінницький національний технічний університет, Вінниця, e-mail: maksm.plis1995@gmail.com;

Maksim V. Plys – student in Computer Control Systems, Vinnytsia National Technical University, Vinnytsia, e-mail: maksm.plys1995@gmail.com;

Oleksiy M. Moskvin – Ph.D., e-mail: moskvin.aleksey@gmail.com;

Svetlana M. Moskvina – Ph.D, Professor of the Chair of Computer Control Systems, Vinnytsia National Technical University, Vinnytsia, email : moskvina@ukr.net.