УДК 004.3

ДОСЛІДЖЕННЯ ПАРАЛЕЛЬНИХ АЛГОРИТМІВ ПОШУКУ ХАРАКТЕРНИХ НАБОРІВ З ВИКОРИСТАННЯМ ТЕХНОЛОГІЙ ПРОГРАММИРОВАНИЯ GPU

К.Ю. Островська, І.А. Бєлих
(Україна, Дніпро, Національна металургійна академія України)

Постановка проблеми. Інтелектуальний аналіз даних є актуальною проблемою в області інформаційних технологій, оскільки ставить найбільш важкі завдання алгоритмічного характеру, які пов'язані з необхідністю забезпечення обчислювальної ефективності, точності, стійкості обробки накопичених знань.

Як правило, завданням інтелектуального аналізу даних є виявлення в сирих даних раніше невідомих, нетривіальних, практично корисних, доступних інтерпретації знань, необхідних для прийняття рішень в різних сферах людської діяльності. При цьому найбільш перевагу надають алгоритми, що використовують мінімальну кількість атрибутів, що безпосередньо пов'язано з їх обчислювальної ефективністю.

Серед найбільш перспективних підходів до вирішення даного завдання є підхід з виявлення асоціативних правил в даних, які відображають не просто статистичні залежності деяких атрибутів, але і причинно - наслідкові зв'язки, що існують в даних. Це дозволяє скоротити розмірність даних, зберігаючи найбільш інформативні атрибути.

Асоціативний аналіз спочатку був спрямований на задоволення потреб маркетингу: аналіз купівельної корзини, прогнозування попиту на товари. В даний час зростає інтерес до його використання в системах ухваленнях рішень для прогнозування громадської думки в соціології, в задачах медичної діагностики, ідентифікації шахрайських операцій на фінансових ринках і в багатьох інших прикладних напрямках.

Пошук асоціативних зв'язків умовно ділиться на два етапи, представлених на рисунку 1.

Рис. 1. Схема пошуку асоціативних правил

На першому етапі генеруються часті набори (frequent item sets), які задовольняють деякій мінімальній підтримки. На другому етапі, на основі частих наборів, здійснюється пошук асоціативних правил в відповідно до мінімального рівня довіри. На думку авторів [1] етап пошуку частих наборів є найбільш трудомістким, тому що вимагає звернення до бази транзакцій, перебору комбінацій значного числа елементів, з метою знаходження задовольняючих заданій підтримці. Рішення, забезпечують ефективне використання обчислювальних ресурсів систем при генерації частих наборів, безсумнівно, є актуальним напрямком досліджень при розробці алгоритмів пошуку асоціативних зв'язків.

В роботі [2] зазначається, що використання паралельної обробки даних на відеочіпах (GPU) дає істотний приріст продуктивності математичних розрахунків. Технологія програмування на відкритих (GPGPU) надає розробнику інструментарій для перенесення неграфічних обчислень на GPU, що дозволяє організовувати паралельне завантаження виконавчих блоків.

Аналіз стану досліджень і розробок алгоритмів пошуку характерних наборів, розроблених з використанням технології GPGPU показав, що існуючі популярні алгоритми, такі як Eclat, FP-Growth мають реалізації для виконання тільки на багатопроцесорних CPU [3].

Дана обставина вимагає проведення досліджень для пошуку альтернативних рішень, що забезпечують паралельне виконання пошуку частих наборів на GPU.

Метою даної роботи є дослідження і розробка паралельних алгоритмів пошуку частих (характерних) наборів даних, що використовуються в завданнях пошуку асоціативних правил, проведення експериментальної оцінки знайдених рішень за критеріями обчислювальної ефективності.

Висновки. У роботі вирішена актуальна задача підвищення ефективності інструментів асоціативного аналізу – досліджені популярні алгоритми пошуку характерних наборів Eclat і FPG і розроблені їх паралельні версії з використанням крос-апаратного і платформного сердовища програмування GPU OpenCL, а також вирішені наступні завдання:

1. Розроблено структуру зберігання транзакцій, що забезпечує перенос алгоритму FPG для виконання на GPU, а також процедури, забезпечують пошук і витяг частих наборів.

2. Досліджено ефективні рішення підрахунку підтримки для алгоритму Eclat з використанням підрахунку в циклі і підрахунку на основі функцій робочих груп, що реалізують паралельно алгоритми scan і reduce.

Аналіз розроблених алгоритмів з використанням реальної бази транзакцій показав наступний результат:

1. Алгоритм Eclat в цілому демонструє кращу тимчасову продуктивність, споживання оперативної пам'яті і завантаження CPU в порівняно з алгоритмом FPG на наборах даних з великою кількістю різнорідних атрибутів, зростання кількості яких призводить до експоненціального підвищення часу генерації наборів для алгоритму FPG проти лінійного росту для алгоритму Eclat. Однак, при збільшенні числа транзакцій відбувається збільшення кількості бітового

уявлення, підрахунок підтримки на якому призводить до лінійного зростання тимчасової продуктивності алгоритму Eclat, тоді як алгоритм FPG залишається нечутливим до даного параметру.

2. Паралельні алгоритми Eclat показали ефективне споживання ресурсів оперативної пам'яті, CPU, а також високу тимчасову продуктивність в порівнянні з послідовним алгоритмом на CPU при числі транзакцій більше 5000, при меншому значенні алгоритми на GPU неефективні через витрати на перемикання контексту host-> device-> host.

3. Підрахунок підтримки для алгоритму Eclat на основі функцій робочих груп, які наявні в стандарті OpenCl 2.0, показав кращу тимчасову ефективність в порівнянні з підрахунком в циклі, що пояснюється оптимізацією операцій додавання всередині робочої групи і відсутності витратних операцій завантаження і вивантаження результатів складання.

4. Паралельні алгоритми FPG показали також ефективне споживання ресурсів оперативної пам'яті, CPU, а також високу тимчасову продуктивність в порівнянні з послідовним алгоритмом на CPU навіть на невеликих наборах даних (починаючи від десятків транзакцій), тому що перемикання контексту, на відміну від алгоритму Eclat, який рекурсивно формує дерево частих наборів і для кожного вузла викликає kernel - функцію, проводиться лише двічі - під час передачі префіксного дерева в kernel-функцію і при вивантаженні результатів з вихідного буфера.

Варто відзначити, що паралельний алгоритм FPG має точку підвищення ефективності використання пам'яті GPU, яка полягає в розробці способів псевдодінамічного виділення пам'яті в kernel - функції. Як відомо, виділення пам'яті для обчислення на GPU пристрої виробляється на стороні host, що не завжди враховує реальну потребу процедур, що виконуються побудова структур, розмір яких заздалегідь не відомий. Одним із способів реалізіації динамічного управління пам'яттю на стороні GPU - пристрою є створення єдиного буфера значного розміру, переданого в kernel - функцію і визначення функцій, аналогічних malloc і free в стандартній бібліотеці C. Однак, визначення не надлишкових розмірів згаданого буфера вимагає проведення додаткових досліджень алгоритму FPG.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Guizhen Yang, The Complexity of Mining Maximal Frequent Item sets and –Maximal Frequent Patterns: [Электронный документ]. (https://www.semanticscholar.org/paper/The-complexity-of-mining-maximal-frequent-itemsets-Yang/6b92231d9815544d9c78891f94d246096b1393c8).

2. С.А. Полетаев, Параллельные вычисления на графических процессорах: [Электронный документ]. // Параллельное программирование. – 2012. – 300 с. (https://www.iis.nsk.su/files/articles/sbor_kas_16_poletaev.pdf).

3. Tianyuan Jiang & Prepost: A GPU Accelerated–Xin Lv, Zhihong Deng, GPU Frequent Pattern Mining Algorithm Based on PrePost: [Электронный документ]. (https://jtyuan.github.io/files/gpu-prepost.pdf).