

Кластеризація та аналіз продуктивності програмного коду

Вінницький національний технічний університет

Анотація:

Кластеризація даних є важливою технікою для аналізу продуктивності програмного коду. Використовуючи методи кластеризації, можна виділити закономірності у продуктивності різних частин коду та виявити області, що потребують оптимізації. У цій роботі розглянуто застосування кластеризації для аналізу результатів тестування продуктивності, зокрема для оцінки ресурсомісткості та часу виконання різних функцій програми. Дослідження підходів, що включають застосування методу *k-means* та ієрархічної кластеризації, дозволяє запропонувати методики для групування функцій за їхніми параметрами продуктивності та оптимізувати код у відповідних групах.

Ключові слова: кластеризація, аналіз продуктивності, оптимізація коду, метод *k-means*, ієрархічна кластеризація, продуктивність програм.

Abstract:

Data clustering is a valuable technique for analyzing software performance. By applying clustering methods, patterns in the efficiency of different code segments can be identified, revealing areas for optimization. This study investigates clustering approaches for analyzing performance test results, particularly in terms of resource usage and execution time of various program functions. Research on *k-means* and hierarchical clustering provides methodologies to group functions based on performance metrics and optimize code within respective clusters.

Keywords: clustering, performance analysis, code optimization, *k-means* method, hierarchical clustering, software efficiency.

Вступ

Кластеризація даних є широко вживаною технікою для виявлення структури в даних, а також може бути корисною для аналізу продуктивності програмного коду. Зі зростанням складності програмних продуктів аналіз продуктивності та виявлення критичних ділянок коду є важливими задачами для покращення ефективності. Дана робота зосереджена на методах кластеризації для групування функцій коду за їхніми характеристиками продуктивності та аналізу продуктивності програмного забезпечення.

Методи та інструменти кластеризації

Алгоритми *k-means* та ієрархічної кластеризації застосовуються для групування функцій за параметрами продуктивності, такими як час виконання, використання пам'яті, частота викликів. Параметри кожної функції зберігаються у вигляді векторів, де значення кожного елемента представляє певний показник продуктивності.

Основною метою методу *k-means* є мінімізація сумарної відстані між об'єктами та центроїдами їхніх кластерів. Для кожної функції обчислюється середнє значення показників продуктивності:

$$J = \sum_{i=1}^k \sum_{x \in C_i} \|x - \mu_i\|^2$$

де:

- x - точка даних, що представляє продуктивність функції;
- C_i - кластер, до якого належить точка;
- μ_i - центроїд кластера C_i ;
- J - функція втрат, яку мінімізують.

Ієрархічна кластеризація використовується для дослідження взаємозв'язків між функціями та їх впливу на продуктивність всієї програми. Ієрархічний підхід дозволяє побудувати деревоподібну структуру кластерів, яка допомагає визначити, які функції потребують спільної оптимізації. Формально, відстань між кластерами A і B визначається як:

$$d(A, B) = \min_{x \in A, y \in B} \|x - y\|$$

Де $d(A, B)$ - відстань між кластерами A і B , що дозволяє об'єднувати кластери з найбільш близькими функціями за характеристиками продуктивності.

Практичний аналіз продуктивності

Для аналізу продуктивності було розроблено програмний код для тестування різних функцій, зокрема обчислення факторіалів, сортування великих масивів та реверсування текстових рядків. Для кожної функції вимірювалися такі показники, як час виконання, використання пам'яті та кількість операцій введення-виведення.

Наприклад, функція сортування масиву мала значно більший час виконання у порівнянні з іншими функціями через обчислювальну складність алгоритму. Використовуючи метод k -means, було створено три кластери: функції з високою продуктивністю, функції із середньою продуктивністю та функції, що потребують оптимізації. Функції сортування масивів і складні обчислення факторіалів потрапили в кластер низької продуктивності, що вказує на необхідність оптимізації, наприклад, заміни алгоритму сортування на більш ефективний або застосування спеціальних структур даних.

Також було проведено додатковий аналіз коду за допомогою профілювальних інструментів для виявлення вузьких місць, що підтвердило результати кластеризації.

Переваги кластеризації для аналізу коду

Кластеризація дозволяє систематизувати великий обсяг даних продуктивності, що спрощує виявлення проблемних ділянок у програмі. Зокрема, метод k -means дозволяє швидко розподілити функції за продуктивністю та виокремити кластери для оптимізації, а ієрархічний підхід дає змогу ідентифікувати більш складні зв'язки між функціями.

Висновки

Кластеризація є ефективним інструментом для аналізу продуктивності програмного коду. Застосування методів k -means та ієрархічної кластеризації дозволяє групувати функції за параметрами продуктивності, що спрощує процес ідентифікації проблемних ділянок коду. Оптимізація на основі отриманих кластерів сприяє зменшенню витрат ресурсів та підвищенню ефективності програмного забезпечення.

На основі аналізу коду, проведеного у цій роботі, вдалося виокремити критичні ділянки, які найбільше впливають на продуктивність програми, і зосередити зусилля оптимізації на найбільш ресурсомістких функціях. Застосування кращих алгоритмів сортування, спеціалізованих структур даних та інших методів дозволило значно скоротити час виконання програми та зменшити навантаження на пам'ять. Результати підтверджують, що кластеризація є потужним інструментом для підвищення продуктивності програмного забезпечення, особливо у великих проектах, де важливо правильно пріоритизувати ресурси для оптимізації.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Слободянюк, В. П. Методологія дослідження програмного забезпечення. – Львів: Вид-во ЛНУ імені Івана Франка, 2017. – 250 с.
2. Бақун, В. Я., Качан, А. А. Основи кластеризації даних. – Одеса: Одеський національний університет, 2015. – 180 с.
3. Волошин, Ю. В. Аналіз та оптимізація програмного забезпечення. – К.: Наукова думка, 2018. – 300 с.
4. Загребельний, М. І. Методи кластеризації та їх застосування. – Запоріжжя: Запорізький національний університет, 2019. – 220 с.
5. Сухомлин, О. В. Продуктивність програмного забезпечення: теорія та практика. – Київ: Видавництво "Академперіодика", 2020. – 260 с.

Сахно Михайло Миколайович – студент групи 2КІ-23б, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, Вінниця, e-mail: sahnomihajlo51@gmail.com

Науковий керівник: **Добровольська Наталія Вікторівна** – доцент кафедри обчислювальної техніки, Вінницький національний технічний університет, м. Вінниця, email: dobr_n_v@vntu.edu.ua

Sakhno Mykhailo M. - students, 2CE-23B, Faculty of information Technologies and Computer Engineering, Vinnytsa National Technical University, email: sahnomihajlo51@gmail.com

Supervisor: **Dobrovolskaya Natalia Viktorivna** - Associate Professor at the Department of Computer Engineering, Vinnytsia National Technical University, Vinnytsia, Ukraine, email: dobr_n_v@vntu.edu.ua