ISSN 2707-1898 (print)
Український журнал інформаційних технологій
Ukrainian Journal of Information Technology
http://science.lpnu.ua/uk/ujit
https://doi.org/10.23939/ujit2022.02.018
Article received 05.10.2022 p.
Article accepted 17.10.2022 p.
UDK 004.891

Correspondence author
Ye.V. Levus
Yevheniia.V. Levus@lpnu.ua

Є. В. Левус, Р. Б. Василюк
Національний університет "Львівська політехніка", м. Львів, Україна

РЕКОМЕНДАЦІЙНИЙ АЛГОРИТМ ІЗ ВИКОРИСТАННЯМ КЛАСТЕРИЗАЦІЇ ДАНИХ

Розроблено рекомендаційний алгоритм для підвищення якості надання рекомендацій та врахування проблем розрідженості даних і холодного старту, який враховує удосконалення відомих методів колаборативної фільтрації із використанням кластеризації. З'ясовано, що рекомендаційні системи створюють для швидкого знаходження товарів чи послуг в мережі Інтернет, надаючи пропозиції, які точно відповідають інтересам користувачів. Незважаючи на інтенсивний розвиток алгоритмів рекомендаційних систем та незалежно від доменів їх використання (електронна комерція, розваги, послуги, новини, соціальні мережі тощо), актуальними залишаються питання підвищення якості наданих рекомендацій, збільшення швидкодії їх вироблення, масштабованості, забезпечення стійкості в умовах розрідженості даних, холодного старту. Модифіковано алгоритм колаборативної фільтрації, який можна використати для вироблення рекомендацій користувачам системами закупівлі товарів чи надання послуг. Виявлено, що завдання пошуку схожих користувачів за вподобанням вирішують з використанням кластеризації. Поділ користувачів на кластери відбувається за допомогою алгоритму *k*-середніх із динамічним пошуком оптимальної кількості кластерів і початкових центроїдів. Запропонований рекомендаційний алгоритм надає релевантні рекомендації та працює ефективно за різної кількості вхідних даних. Кластеризація дає змогу алгоритму бути масштабованим і працювати із великою кількістю користувачів системи. Практична реалізація модифікованого рекомендаційного алгоритму здійснена для системи підбору кінофільмів. Наукова новизна отриманих результатів дослідження полягає у розвитку методу колаборативної фільтрації на підставі використання кластеризації із динамічним визначенням кількості кластерів і початкових центроїдів для ідентифікації груп подібних користувачів. Для верифікації результатів модифікований алгоритм було порівняно із іншими наявними імплементаціями – з алгоритмом, заснованим на пам'яті, та алгоритмом, заснованим на сусідстві. Запропонований алгоритм має кращі результати на 25-40 % для проведених тестів. Модифікований рекомендаційний алгоритм не є прив'язаним до певної предметної області, тому його можна інтегрувати в програмні системи різних доменів.

Ключові слова: центроїди; кластери; коефіцієнт подібності; розрідженість даних; холодний старт.

Вступ/Introduction

Вибір того чи іншого товару або послуги серед величезної кількості пропозицій від численних онлайн сервісів, платформ, що найбільше відповідає інтересам клієнта, є дуже складним, ресурсномістким і заплутаним завданням. Полегшити вирішення такого завдання можливо через використання рекомендаційних алгоритмів, потреба в яких стає все відчутнішою для забезпечення прибутковості бізнесу. Якісно сформовані рекомендації дають змогу підвищити інтерес клієнтів до товару чи послуги, у зв'язку з цим більшість підприємців, які займаються онлайн продажем певних товарів чи послуг, все частіше бажають імплементувати рекомендаційні системи у власні сервіси.

Яскравим прикладом сервісу, який використовує один з популярних рекомендаційних алгоритмів – колаборативну фільтрацію для підвищення зацікавленості користувача, є компанія Netflix, яка позиціонує себе як стрімінговий онлайн-кінотеатр. Головним принципом його роботи є аналіз вподобань користувачів і створення на його основі відповідних рекомендацій. Колаборативна фільтрація дає змогу знайти та проаналізувати приховані фактори та надати неочікувані рекомендації, які зрештою будуть успішними [3], [9].

Пошук подібних користувачів є одним із найбільш важливих завдань рекомендаційних систем. Подібність в кожній системі визначають за допомогою даних, що система зібрала про користувачів [12]. Такими даними можуть бути інформація про переглянуті елементи, наданий рейтинг, відгуки користувача, переходи за посиланнями, інформація з профілю користувача, яка була надана під час реєстрації в системі тощо.

Незважаючи на інтенсивний розвиток теорії рекомендаційних алгоритмів і велику кількість їх реалізацій, багато питань залишаються невирішеними [11]. Підвищенню якості сформованих рекомендацій присвячена велика кількість наукових робіт та досліджень [7], [13], [16], [22]. Незалежно від доменів, у яких працюють рекомендаційні системи (торгівля, сфера розваг, новини тощо) виділяють низку проблем, з якими вони зіштовхуються. Передусім, це труднощі в забезпеченні релевантності рекомендацій, зниження ефективності рекомендаційних систем із зростанням кількості користувачів і рекомендаційних елементів, проблема холодного старту в нових користувачів системи, урахування розрідженості даних.

Об'єкт дослідження – надання рекомендацій користувачам із підбору товару чи послуги.

Предмет дослідження – модифікація алгоритму колаборативної фільтрації із використанням кластеризації для пошуку подібності користувачів, зацікавлених, наприклад, в підбору кінофільмів.

Мета роботи – підвищення якості надання рекомендації та врахування проблем розрідженості даних і холодного старту, який враховує удосконалення відомих методів колаборативної фільтрації із використанням кластеризації.

Для досягнення зазначеної мети визначено такі *основні завдання дослідження*:

- проаналізувати наявні рекомендаційні рішення щодо використання кластеризації даних для класифікації користувачів;
- модифікувати алгоритм кластеризації, додавши динамічне обчислення необхідних параметрів для кластеризації, які істотно впливають на остаточний результат рекомендування;
- розробити рекомендаційний алгоритм, що шукає подібність між користувачами одного кластеру із можливістю створення переліку рекомендацій, що базується на рейтингу, який користувач надав певним рекомендаційним елементам системи;
- програмно імплементувати розроблений алгоритм та колаборативні алгоритми, засновані на пам'яті та на сусідстві, для системи підбору кінофільмів;
- проаналізувати отримані результати та порівняти набори отриманих рекомендацій із результатами інших рекомендаційних алгоритмів;
- розглянути результати роботи розробленого алгоритму у випадку розрідженості даних і холодного старту.

Аналіз останніх досліджень та публікацій. У сучасних рекомендаційних системах існує безліч підходів до пошуку рекомендацій, деякі з них базуються на фільтрації вмісту, а інші – на колаборативній фільтрації. Історично першим рекомендаційним алгоритмом вважається алгоритм фільтрації вмісту. Він дає змогу аналізувати подібні рекомендаційні елементи і, базуючись на тому чи елемент був релевантним для користувача, пропонує схожі елементи із переліку. Цей тип алгоритмів є простішим відносно інших, проте недоліком є те, що різноманіття рекомендацій не є високим, оскільки використовують внутрішню модель пошуку схожих елементів за критеріями [2], [11], [16].

Алгоритми із використанням колаборативної фільтрації пропонують більш точніші рекомендаційні елементи ніж фільтрація вмісту, проте використовують складніші обчислення. Дана точність досягається за рахунок того, що під час фільтрації вмісту аналізуються дії тільки одного певного користувача, історія його пошуку, вподобання, візити на той чи інший портал тощо. Цей алгоритм обробляє дані про користувача і генерує пропозицію із переліку власних елементів [2].

Найпоширенішими методами у рекомендаційних системах для підвищення якості рекомендацій та мінімізації негативних впливів відомих проблем, зокрема, розрідженості даних і холодного старту, вважаються гібридизація методів колаборативної фільтрації, рейтингування усіх користувачів, машинне навчання, кластеризація даних [8], [10], [14], [19].

Незважаючи на те, що методи кластеризації часто використовують у задачах інтелектуального аналізу даних в комерційних і наукових сферах і як інструмент візуалізації даних, у літературі бракує вторинних досліджень, які б аналізували використання алгоритмів кластеризації в рекомендаційних системах та їх поведінку в різних аспектах [5], [17], [15].

Кластеризацію вважають одним з методів машинного навчання, а саме спонтанного навчання або навчання без учителя. Такий підхід не вимагає попереднього навчання моделі на різних наборах даних і втручання розробника у сам процес вироблення рекомендацій чи прогнозів, що значно спрощує імплементацію рекомендаційних систем [1], [20].

Популярним типом алгоритмів кластеризації є ті, що засновані на центроїдних моделях, серед них – алгоритм *k*-середніх. Це ітераційні алгоритми кластеризації, в яких поняття подібності випливає з близькості точки даних до центроїда власного кластеру. Недоліком алгоритмів даного типу є те, що кількість кластерів, на яку необхідно розбити вхідну множину даних, повинна бути вказана до початку виконання самого алгоритму [21]. Це вимагає попереднього аналізу вхідного набору даних для визначення кількості кластерів.

Ще однією особливістю кластеризації методом *k*-середніх є випадкова ініціалізація початкових центроїдів, що є швидким, проте не надто надійним рішенням. Перевагами даного підходу, що описаний в оригінальній імплементації алгоритму *k*-середніх, є те, що немає необхідності проводити додаткові обчислення. Проте найбільшим недоліком є висока імовірність вибору усіх центроїдів занадто близько один до одного. Це може привести до неправильної кластеризації елементів, роблячи деякі кластери перевантаженими. Важливо те, що під час випадкової ініціалізації неможливо передбачити чи центроїди будуть розташовані коректно [4], [21].

Огляд публікацій показав, що актуальним завданням є розвиток відомих методів колоборативної фільтрації на основ використання методу кластеризації. Ефективне використання методу кластеризації у рекомендаційних ситемах вимагає вдосконалення пошуків кількості кластерів і початкових значень центроїдів.

Результати дослідження та їх обговорення/Research results and their discussion

Ефективність рекомендаційної системи можна підвищити, якщо алгоритми вироблення рекомендацій оперуватимуть відразу із групами подібних між собою елементів. Для класифікації таких груп проводять кластерний аналіз. Його головним завданням є поділ певної вибірки на підмножини – кластери, усі елементи в межах одного з них будуть схожими, а елементи різних кластерів відрізнятимуться між собою. Даний підхід підвищує швидкодію системи та точність рекомендацій завдяки тому, що пошук безпосередньо рекомендацій проводитиметься у межах одного кластеру, усі елементи якого і так уже є подібними.

Методом кластеризації для реалізації рішення щодо підвищення якості рекомендацій обрано метод *k*-середніх, зважаючи на його швидкість, гнучкість, можливості до масштабування та простоту в імплементації та модифікації. Оскільки даний алгоритм є ітераційним, то у власній модифікації даного алгоритму можна обмежити максимально допустимо кількість ітерацій задля підвищення швидкодії та уникнення зациклень системи чи витоків пам'яті [4]. Незважаючи на швидкість, у даного методу є ряд недоліків. Стабільність алгоритму можна підвищити у випадках автоматизованого вибору кількості кластерів і покращеного вибору початкових центроїдів. Оскільки ці модифікації щодо кластерного аналізу зможуть істотно збільшити передбачуваність вмісту майбутніх кластерів відповідно до різних наборів даних, отримаємо формування покращеного остаточного результату.

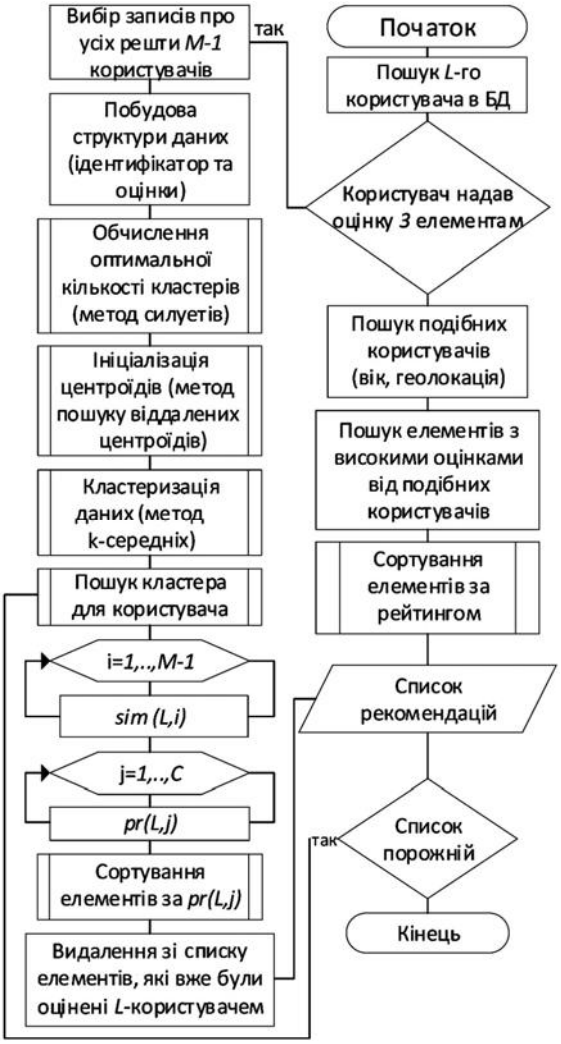

Рис. 1. Блок-схема рекомендаційного алгоритму / Block diagram of the recommendation algorithm

Основними етапи обчислень в рекомендаційному алгоритмі (рис. 1) є:

1) *Ініціалізація даних.* Виконується запит про те, що певному користувачеві необхідно надати рекомендації, за ідентифікатором цього користувача отримують інформацію про оцінки фільмам, які користувач надав раніше.

2) *Випадок рекомендування для нового користувача.* Відповідно від того, чи користувач є новим для системи, алгоритм готує різні дані для рекомендування. Якщо користувач є новим для системи, тобто таким, який ще не оцінив хоча б три фільми, то отримують перелік найбільш популярних фільмів із найвищою оцінкою і робота алгоритму припиняється. Якщо ж користувач оцінив три або більше фільмів, система підбере особисті рекомендації для даного користувача.

3) *Класифікація користувача.* Для початку система повинна класифікувати поточного користувача до певного кластеру даних для того, щоб надати йому найбільш релевантні рекомендації. Кластеризація відбувається за допомогою алгоритму *k*-середніх із динамічним пошуком кількості кластерів і динамічною ініціалізацією початкових центроїдів. Загалом даний етап поділяється на три послідовні процеси, внаслідок яких система зможе оперувати кластеризованими даними.

3.1) Передусім визначають кількість кластерів, на які необхідно поділити набір даних. Для цього використовують спочатку звичайний метод *k*-середніх, а потім – метод силуетів на проміжку від 1 до $\sqrt{n}$, де n – кількість елементів у наборі даних. Цей метод обчислює силуетні коефіцієнти $s(i)$ кожного елементу множини $i \in C_i$, які вимірюють, наскільки кожен із елементів схожий на власний кластер порівняно з іншими кластерами [6], [18]:

$$s(i) = \begin{cases} 1 - \dfrac{a(i)}{m(i)}, & \text{якщо } a(i) < m(i), \\ 0, & a(i) = m(i), \\ \dfrac{a(i)}{m(i)} - 1, & \text{інакше}, \end{cases} \quad (1)$$

де: $a(i)$ – середня відстань між i-ою та іншими точками кластеру (2); $m(i)$ – найменша відстань від i-ої до сусіднього кластеру (3).

Якщо визначити $|C_i|$ – кількість точок, що належить кластеру C_i, а $d(i,j)$ – відстань між точками i та j, то можна обчислити значення:

$$a(i) = \frac{1}{|C_i| - 1} \sum_{j \in C, i \neq j} d(i,j), \quad (2)$$

$$m(i) = \min_{j \neq i} \frac{1}{|C_j|} \sum_{j \in C} d(i,j). \quad (3)$$

За силуетними коефіцієнтами $s(i) \in [-1;1]$ визначають якість групування кластеру: близькість до 1 означає, що кластер добре згрупований, до -1 – приналежність до сусіднього кластеру, 0 – нейтральність елемента.

Здійснюють пошук найбільшого значення коефіцієнту силуету. Оптимальною кількістю кластерів буде k із найвищим коефіцієнтом силуету.

3.2) Якщо оптимальна кількість кластерів відома, то здійснюють пошук початкових центроїдів за допомогою методу пошуку віддалених елементів. Спочатку виконують випадковий вибір одного елементу з множини та здійснюють ініціалізацію першого центроїду в даній точці. Обчислюють відстані від центроїдів до усіх точок множини, що не є центроїдами. Новий центроїд ініціалізують в точці, що розміщена найдальше від усіх раніше визначених центроїдів. Такі обчислення повторюють k разів. Результатом виконання цього кроку є набір із k центроїдів.

3.3) Розпочинають безпосередню кластеризацію за допомогою алгоритму *k*-середніх на даному наборі даних із кількістю кластерів k та початковими центроїдами, які були визначені у попередніх кроках. Результатом цього кроку є сформовані k кластери.

4) *Обчислення коефіцієнтів подібності та вподобання.* У межах сформованого кластеру, в якому знаходиться користувач, якому необхідно надати рекомендації, вираховується коефіцієнт подібності між поточним користувачем та усіма іншими як косинус подібності:

$$sim(A,B) = \frac{\vec{A} \cdot \vec{B}}{\|A\| \cdot \|B\|} = \frac{\sum_{i=1}^{n} A_i \cdot B_i}{\sqrt{A_i^2} \cdot \sqrt{B_i^2}}. \quad (4)$$

Компонентами векторів є оцінки виставлені елементам (товарам чи послугам) користувачами A і B. Для практичної реалізації як A_i, B_i обрано усереднені оцінки жанрам кінофільмів (комедія, пригоди, історичний, сімейний, анімаційний, екшн). Дана середня оцінка оновлюється із кожною новою оцінкою користувача на

підставі обчислень середнього арифметичного для кожного із жанрів.

Для кожного фільму, що був оцінений хоча б одним користувачем із даного кластеру, обчислюють коефіцієнт вподобання фільму

$$pr(L,K) = \frac{1}{M-1} \sum_{N=1, N \neq L}^{M} r_N^{(K)} \cdot sim(L,N), \quad (5)$$

де $r_N^{(K)}$ – оцінка, яку користувач N надав фільму K.

Фільми сортуються у порядку спадання за коефіцієнтом вподобання. Результатом цього кроку є відсортований перелік рекомендаційних елементів.

5) *Формування переліку рекомендацій*. Відбувається перевірка, чи користувач уже оцінив фільми із відсортованого переліку, який був результатом попереднього кроку. Якщо користувач оцінив елемент, фільм не пропонують користувачеві. Сформований перелік рекомендацій надсилається користувачеві.

У випадку, якщо рекомендаційних елементів недостатньо для задоволення запиту (користувач уже оцінив більшість фільмів), відбувається пошук найближчого кластера. Виконуються кроки 4, 5.

Побудований рекомендаційний алгоритм реалізовано окремою компонентою в програмній системі підбору кінофільмів (рис. 2). Програмна система створена за клієнт-серверною архітектурою, де клієнтом є мобільний додаток, створений за допомогою технології React Native, а серверна частина – із використанням технології Node.js. Для доступу до бази даних використано Mongoose ODM, утиліту, яка дає змогу зручно доступатись до даних в MongoDB. Дані із категорій "Фільми" та "Серіали" не є обов'язковими для зберігання безпосередньо в дану базу, оскільки вони зберігаються у сторонній базі даних на ресурсі The MovieDB.

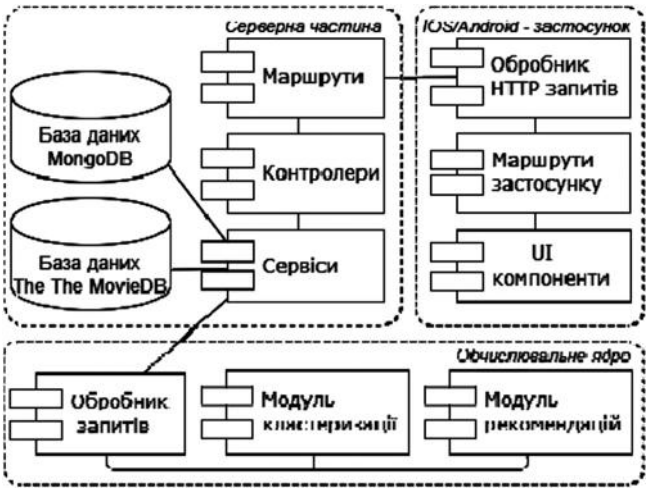

Рис. 2. Архітектура програмної системи / Software system architecture

Для верифікації розроблених алгоритмів проведено низку обчислювальних експериментів. Проаналізовано доцільність модифікованого методу ініціалізації центроїдів на підставі експериментів для набору даних із 100 користувачів. Для визначення якості ініціалізації центроїдів із даного набору даних створено еталон з 5 кластерів. Частка збігу результатів кластеризації із еталоном вказує на ефективність методу. Також виявлено вплив кількості ітерацій на стійкість кластерів (табл. 1).

Згідно із отриманими результатами видно, що випадкова ініціалізація не є оптимальним рішенням, адже результати кластеризації можуть відрізнятися у випадку одних й тих самих вхідних даних. Модифікована ініціалізація дає більш стабільний результат та високу схожість із еталонними кластерами.

Табл. 1. Результати порівняння методів вибору початкових центроїдів / The results of the comparison of the initial centroid selection methods

	Випадкова ініціалізація	Ініціалізація з модифікацією
Експеримент №1		
Кількість ітерацій	6	5
Схожість з еталоном	71.2 %	96.9 %
Експеримент №2		
Кількість ітерацій	7	6
Схожість з еталоном	91.7 %	97.2 %
Експеримент №3		
Кількість ітерацій	7	5
Схожість з еталоном	83.5 %	96.8 %

Для перевірки остаточних результатів рекомендування модифікований алгоритм порівняно з іншими популярними алгоритмами: алгоритмом, що є заснований на пам'яті та алгоритмом, заснованим на сусідстві. Для проведення обчислювального експерименту випадково обрано наявного користувача системи. Суть експерименту полягала в тому, щоб дати змогу алгоритмам порекомендувати фільми даному користувачеві ще раз із обмеженого переліку фільмів, що попередньо були оцінені даним користувачем. Після цього весь перелік рекомендацій відсортували та співставили із оригінальними оцінками користувача. Тобто, якщо користувач поставив 5 "зірок" 20 фільмам, а внаслідок тестування алгоритму фільм, який мав би бути в першій двадцятці, має порядковий номер у остаточному переліку більший за 20, то рекомендація для даного фільму не є успішною і показник якості буде нижчим. Таке порівняння проведено для усіх елементів переліку із рекомендованими фільмами відносно переліку із оригінальними оцінками користувача. Внаслідок чого сформовано остаточну частку успішних рекомендацій. Кожен із алгоритмів повертає перелік із власними коефіцієнтами потенційного вподобання для кожного із фільмів, який сортують та порівнюють із оригінальними оцінками.

Як тестових даних взято три набори – на 100, 500 та 2500 користувачів, кожен з яких відповідно сформував власні рейтинги для різних фільмів. Отримано результати, що свідчать про якісніші результати для модифікованого алгоритму (табл. 2).

Табл. 2. Результати порівняння колаборативних рекомендаційних алгоритмів / Results of the comparison of collaborative recommendation algorithms

№	Алгоритм	Кількість якісних рекомендацій (%) за різної кількості користувачів (N)					
		N=100		N=500		N=2500	
1	Заснований на пам'яті	52,4		48,7		39,6	
2	Заснований на сусідстві	61,2		57,9		42,9	
3	Модифікований алгоритм	87,4		84,1		79,9	
Підвищення якості рекомендування (%)		1 і 3	2 і 3	1 і 3	2 і 3	1 і 3	2 і 3
		35	26,2	35,4	26,2	40,3	37,0

Алгоритми перевірено також на стійкість до відомих проблем колаборативних алгоритмів, таких як проблема холодного старту та проблема розрідженості даних (рис. 3).

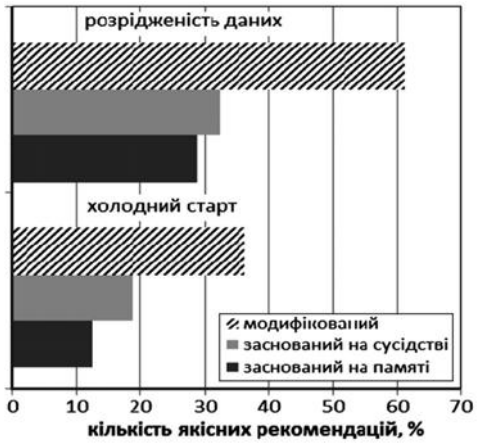

Рис. 3. Результати порівняння колоборативних алгоритмів за умов холодного старту та розрідженості даних/ Results of the comparison of collaborative algorithms under the conditions of cold start and sparse data

Для кожної із проблем було підготовлено відповідні набори даних. Проблема холодного старту симулювалась тим, що користувач, якому необхідно надати рекомендації, не надав жодної оцінки. Для підсилення розрідженості даних було видалено близько 40 % наданих оцінок різними користувачами. Обидва набори даних містили записи про 500 користувачів системи.

Обговорення результатів дослідження. Загалом якість рекомендацій, отриманих модифікованим алгоритмом, становить 35-80 %. Особливо відчутна перевага модифікованого алгоритму у випадку великої кількості користувачів, що підтверджує доцільність застосування кластеризації та динамічного вибору кількості кластерів для кожного окремого набору даних. Падіння якості рекомендацій для модифікованого алгоритму становить менше 10 % при зростанні користувачів у 25 разів (від 100 до 2500). Завдання зменшити негативні впливи проблем рекомендаційних алгоритмів на якість результатів частково вирішено в модифікованому алгоритмі завдяки використанню кластеризації з динамічними визначеннями кількості кластерів і значень початкових центроїдів. Для вирішення проблеми холодного старту важливо те, що в модифікованому алгоритмі пошук проводять серед схожих користувачів системи, базуючись на інформації, яку користувач надав під час реєстрації. Високий рівень якості рекомендацій в умовах високої розрідженості даних зумовлюється тим, що система оперує векторами як середніх рейтингових показників. Проведення операцій у межах одного кластеру, елементи якого і так є схожими один на одного, також спрощує пошук рекомендацій при розрідженості даних.

Отримані результати дослідження дають змогу сформулювати наукову новизну та практичну значущість проведеного дослідження.

Наукова новизна отриманих результатів дослідження – отримав подальший розвиток метод колаборативної фільтрації на підставі використання кластеризації із динамічним визначенням кількості кластерів і початкових центроїдів для ідентифікації груп подібних користувачів.

Практична значущість результатів дослідження – створена рекомендаційна система, яка популяризує надання послуг з перегляду фільмів, а також побудовані алгоритми можна використовувати в рейтингових рекомендаційних системах із можливістю обчислення усереднених оцінок для певних атрибутів. Рекомендаційний алгоритм не є прив'язаним до даної предметної області, тому його можна інтегрувати в інші програмні системи.

Висновки/Conclusions

Розроблено рекомендаційний алгоритм для підвищення якості надання рекомендацій та врахування проблем розрідженості даних, холодного старту на підставі розвитку відомих методів колаборативної фільтрації із використанням кластеризації. За результатами дослідження можна зробити такі основні висновки.

1. Актуальним дослідженням є розвиток рекомендаційних алгоритмів для підвищити якість отриманих результатів, зокрема, у випадках холодного старту та розрідженості даних. У роботі описано розроблений рекомендаційний алгоритм на підставі колоборативної фільтрації з використанням кластеризації. Особливістю є те, що використано динамічні визначення кількості кластерів методом силуетів і значень початкових центроїдів.

2. Рішення програмно імплементовано у вигляді системи рекомендування кінофільмів і верифіковано на підставі порівняння з іншими колоборативними алгоритмами. Якість рекомендацій, отриманих розробленим алгоритмом, є вищою на 25-40 % порівняно з альтернативними алгоритмами. Розроблений рекомендаційний алгоритм надає релевантні рекомендації (якість близько 80 %) та працює ефективно за різної кількості вхідних даних. Кластеризація дає змогу алгоритму бути масштабованим і працювати із великою кількістю користувачів системи.

3. Отримані результати засвідчують перспективність досліджень та вказують на потребу подальшого опрацювання випадку холодного старту. Якість у цьому випадку, хоч і була вищою порівняно з іншими колоборативними алгоритмами, проте становила менше 40 %. Розроблений алгоритм можна використати для вироблення рекомендацій у різних предметних областях.

References

[1] Ahuja, R., Chug, A., Gupta, S., Ahuja, P., & Kohli, S. (2020). Classification and Clustering Algorithms of Machine Learning with their Applications. In: Yang, XS., He, XS. (eds) Nature-Inspired Computation in Data Mining and Machine Learning. Studies in Computational Intelligence, 855. Springer, Cham. https://doi.org/10.1007/978-3-030-28553-1_11

[2] Bansal, S., & Baliyan, N. (2019). A Study of Recent Recommender System Techniques. International Journal of Knowledge and Systems Science (IJKSS), *10*(2), 13-41. http://doi.org/10.4018/IJKSS.2019040102

[3] Brinton, C., & Chiang, M. (2019). Netflix Recommendation System. Retrieved from: https://www.coursera.org/lecture/networks-illustrated/netflix-recommendation-system-TYOZV

[4] Capóa, M., Péreza, A., & Lozano, J. A. (2017). An efficient approximation to the K-means clustering for massive data. *Knowledge-Based Systems*, *117*, 56-69. https://doi.org/10.1016/j.knosys.2016.06.031

[5] Das, J., Mukherjee, P., Majumder, S., & Gupta, Pr. (2014). Clustering-Based Recommender System Using Principles of Voting Theory. *Proceedings of 2014 International Conference on Contemporary Computing and Informatics*, IC3I 2014. http://doi.org/10.1109/IC3I.2014.7019655
[6] Dinh, D. T., Fujinami, T., & Huynh, V. N. (2019). Estimating the Optimal Number of Clusters in Categorical Data Clustering by Silhouette Coefficient. In: Chen, J., Huynh, V., Nguyen, GN., Tang, X. (Eds) Knowledge and Systems Sciences. KSS 2019. *Communications in Computer and Information Science, 1103*. Springer, Singapore. https://doi.org/10.1007/978-981-15-1209-4_1
[7] Gope, J., & Jain, S. K. (2017). A survey on solving cold start problem in recommender systems. 2017 International Conference on Computing. *Communication and Automation* (ICCCA). https://doi.org/10.1109/ ccaa.2017.8229786
[8] Hongzhi, Y., Qinyong, W., Kai, Zh., Zhixu, Li, & Xiaofang, Zh. (2019). Overcoming Data Sparsity in Group Recommendation. IEEE Transactions On Knowledge And Data Engineering, 87-120.
[9] How Netflixs Recommendations System Works. Retrieved from: https://help.netflix.com/uk/node/100639
[10] Huang, Z., Chung, W., & Chen, H. (2003). A Graph Model for E-Commerce Recommender Systems. *Journal of the American Society for Information Science & Technology*, 3–21. https://doi.org/10.1002/asi.10372
[11] Ko, H., Lee, S., Park, Y., & Choi, A. (2022) A Survey of Recommendation Systems: Recommendation Models, Techniques, and Application Fields. *Electronics*, 22(11), 141. https://doi.org/10.3390/electronics11010141
[12] Koutrika G. (2018). Modern Recommender Systems: from Computing Matrices to Thinking with Neurons. Proceedings of the Management of Data, 1651–1654. https://doi.org/10.1145/3183713.3197389
[13] Levus, Ye. V., & Polianska, A. O. (2020). Algorithm for developing a complex recommendation through the example of the tourism industry. *Scientific Bulletin of UNFU*, *30*(5), 122–127. https://doi.org/10.36930/40300520
[14] Lika, B., Kolomvatsos, K., & Hadjiefthymiades, St. (2014). Facing the cold start problem in recommender systems. Expert Systems with Applications, 2065–2073. https://doi.org/10.1016/j.eswa.2013.09.005
[15] Lobur, M., Shvarts, M., & Stekh, Y. (2018). Application of recommender systems in the design of complex microsystem devices. *International Journal of Advanced Research in Computer Engineering & Technology*, *7*(9), 709–714.
[16] Lytvyn, V., Vysotska, V., Shatskykh, V., Kohut, I., Petruchenko, O., Dzyubyk, L., Bobrivetc, V., Panasyuk, V., Sachenko, S., & Komar, M. (2019). Design of a recommendation system based on collaborative filtering and machine learning considering personal needs of the user. *Eastern-European Journal of Enterprise Technologies*, 4(2)(100), 6–28. https://doi.org/10.15587/1729-4061.2019.175507
[17] Miranda, L., Viterbo, J., Bernardini, F. (2020). Towards the Use of Clustering Algorithms in Recommender Systems. AMCIS 2020 Proceedings. 21. Retrieved from: https://aisel.aisnet.org/amcis2020/ai_semantic_for_intelligent_info_systems/ai_semantic_for_intelligent_info_systems/21
[18] Rousseeuw, P. J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics, 20*, 53–65. https://doi.org/10.1016/0377-0427(87)90125-7
[19] Schwarz, M., Lobur, M., Stekh, Y. (2017). Analysis of the effectiveness of similarity measures for recommender systems. The experience of designing and application of CAD systems in microelectronics (CADSM): proceedings 14th International conference, 275-277.
[20] Shutaywi, M., & Kachouie, N. N. (2021). Silhouette Analysis for Performance Evaluation in Machine Learning with Applications to Clustering. *Entropy*, *23*, 759. https://doi.org/10.3390/e23060759
[21] Syakur1, M. A., Khotimah1, B. K., Rochman1, E. M., & Satoto, B. D. (2018). Integration K-Means Clustering Method and Elbow Method For Identification of the Best Customer Profile Cluster. IOP Conference Series Materials Science and Engineering, 2–5.
[22] Zhang, Q., Lu, J., & Jin, Y. (2021). Artificial intelligence in recommender systems. Complex Intell. Syst., *7*, 439–457. https://doi.org/10.1007/s40747-020-00212-w

Ye. V. Levus, R. B. Vasyliuk

Lviv Polytechnic National University, Lviv, Ukraine

RECOMMENDATION ALGORITHM USING DATA CLUSTERING

Recommender systems play a vital role in the marketing of various goods and services. Despite the intensive growth of the theory of recommendation algorithms and a large number of their implementations, many issues remain unresolved; in particular, scalability, quality of recommendations in conditions of sparse data, and cold start. A modified collaborative filtering algorithm based on data clustering with the dynamic determination of the number of clusters and initial centroids has been developed. Data clustering is performed using the k-means method and is applied to group similar users aimed at increase of the quality of the recommendation results. The number of clusters is calculated dynamically using the silhouette method, the determination of the initial centroids is not random, but relies on the number of clusters. This approach increases the performance of the recommender system and increases the accuracy of recommendations since the search for recommendations will be carried out within one cluster where all elements are already similar. Recommendation algorithms are software-implemented for the movie recommendation system. The software implementation of various methods that allow the user to receive a recommendation for a movie meeting their preferences is carried out: a modified algorithm, memory and neighborhood-based collaborative filtering methods. The results obtained for input data of 100, 500 and 2500 users under typical conditions, data sparsity and cold start were analyzed. The modified algorithm shows the best results – from 35 to 80 percent of recommendations that meet the user's expectations. The drop in the quality of recommendations for the modified algorithm is less than 10 per cent when the number of users increases from 100 to 2500, which indicates a good level of scalability of the developed solution. In the case of sparse data (40 percent of information is missing), the quality of recommendations is 60 percent. A low quality (35 percent) of recommendations was obtained in the case of a cold start – this case needs further investigation. Constructed algorithms can be used in rating recommender systems with the ability to calculate averaged scores for certain attributes. The modified recommendation algorithm is not tied to this subject area and can be integrated into other software systems.

Keywords: similarity coefficient; centroid; cluster; data sparsity; cold start.

Інформація про авторів:

Левус Євгенія Василівна, канд. техн. наук, доцент, кафедра програмного забезпечення.
Email: Yevheniia.V.Levus@lpnu.ua; https://orcid.org/0000-0001-5109-7533

Василюк Ростислав Богданович, магістр, кафедра програмного забезпечення. **Email:** rostyslaw.wasyliuk@gmail.com

Цитування за ДСТУ: Левус Є. В., Василюк Р. Б. Рекомендаційний алгоритм із використанням кластеризації даних. *Український журнал інформаційних технологій.* 2022, т. 4, № 2. С. 18–24.

Citation APA: Levus, Ye. V., & Vasyliuk, R. B. (2022). Recommendation algorithm using data clustering. *Ukrainian Journal of Information Technology, 4*(2), 18–24. https://doi.org/10.23939/ujit2022.02.018