ISSN 2707-1898 (print)

Український журнал інформаційних технологій
Ukrainian Journal of Information Technology
http://science.lpnu.ua/uk/ujit
https://doi.org/10.23939/ujit2020.02.043
Article received 13.10.2020 p.
Article accepted 25.10.2020 p.
UDK 004.93

Correspondence author
N. I. Melnykova
melnykovanatalia@gmail.com

Н. Б. Шаховська, Н. І. Мельникова

Національний університет "Львівська політехніка", м. Львів, Україна

МЕТОДИ ПОБУДОВИ МОДЕЛІ ПОВЕДІНКИ КОРИСТУВАЧІВ

Наведено методи побудови моделі поведінки користувачів, які дадуть змогу виявити закономірності планування зустрічей друзів на підставі аналізу їхнього щоденного руху. Для цього попередньо проаналізовано низку методів і алгоритмів кластеризації даних і виокремлено особливості їхнього застосування. З'ясовано, що основними перевагами методів кластеризації даних на підставі їхньої щільності є можливість виявлення кластерів вільної форми різного розміру та стійкості до шуму та викидів. Однак до недоліків цих методів можна віднести високу чутливість до встановлення вхідних параметрів, не чіткий опис класів і непридатність для кластеризації даних великих розмірів. З'ясовано, що основною проблемою всіх алгоритмів кластеризації є їх масштабованість із збільшенням обсягу оброблених даних. Встановлено, що основними проблемами більшості з них є складність налаштування оптимальних вхідних параметрів (для алгоритмів щільності, сітки чи моделі), ідентифікація кластерів різної форми та щільності (алгоритми розподілу, алгоритми на підставі сітки), нечіткі критерії завершення (ієрархічний, розділовий та на підставі моделі). Оскільки процедура кластеризації є тільки одним із етапів оброблення даних системи загалом, обраний алгоритм повинен бути простим у використанні та простим для налаштування вхідних параметрів. Дослідження показують, що ієрархічні методи кластеризації містять ряд алгоритмів, придатних як для оброблення даних невеликого обсягу, так і для аналізу великих даних, що є актуальним у галузі соціальних мереж. На підставі виконаного аналізу даних, зібрано інформацію для заповнення розумного профілю користувача. Значну увагу приділено дослідженню асоціативних правил, на підставі чого запропоновано алгоритм для вилучення асоціативних правил, що дало змогу знаходити статистично значущі правила, а також шукати тільки залежності, визначені загальним набором вхідних даних, та має високу обчислювальну складність, якщо існує багато правил класифікації. Розроблено підхід, що орієнтований на створення та розуміння моделей поведінки користувачів, прогнозування майбутньої поведінки за допомогою створеного шаблону. Досліджено методи моделювання попереднього оброблення даних (кластеризація) та виявлено закономірності планування зустрічей друзів на підставі аналізу щоденного руху людей та їхніх друзів. Наведено методи створення та розуміння моделей поведінки користувачів, застосовано алгоритм *k*-means для групування користувачів, що дало змогу визначити, наскільки добре кожен об'єкт знаходиться у своєму кластері. Введено поняття правил асоціації, розроблено метод пошуку залежностей, оцінено точність моделі.

Ключові слова: вибірка шаблонів, послідовний асоціативний аналіз, кластеризація.

Вступ

Відомо [4], штучний інтелект AI (англ. *Artificial intelligence*) – розділ комп'ютерної лінгвістики та інформатики, що опікується формалізацією проблем і завдань, які подібні до дій, що виконує людина. Це здатність інженерної системи (англ. *Engineered System*) здобувати, обробляти та застосовувати знання та вміння. Останніми роками багато країн виділяють колосальні суми на розвиток штучного інтелекту. Так, зовсім недавно з'явилася інформація про те, що адміністрація президента США Дональда Трампа планує звернутися до Конгресу із запитом про збільшення бюджетних витрат на штучний інтелект і квантові дослідження вдвічі. Проблема в тому, що штучний інтелект може підвищити рівень національної безпеки, освіти та економіки.

Поряд зі штучним інтелектом останнім часом стали відомі такі терміни, як машинне навчання і видобуток даних (англ. *Data Mining*). Основне завдання цих технологій у бізнесі – навчитися розуміти поведінку споживача в умовах, що постійно змінюються. Також буде відомо, як поводитиметься клієнт у майбутньому, позаяк це дасть змогу менеджерам найкраще планувати та здійснювати маркетингові заходи.

Аналіз спорідненості є одним із найпоширеніших методів аналізу даних. Метою цього методу є дослідження взаємозв'язку між подіями, що відбуваються разом. Аналіз спорідненості різновидів – це аналіз кошика ринку, ідентифікація асоціацій між різними подіями, послідовний пошук тощо. Отже, завдання аналізу спорідненості полягає у пошуку правил кількісного опису взаємозв'язку двох або більше подій. Такі правила називають правилами асоціації [11]. Зазвичай, такі асоціативні правила потребують визначення попередньої структури даних, наприклад, групування даних за певними критеріями [12].

Зважаючи на наявні методи та алгоритми машинного навчання, а також враховуючи їхні переваги та недоліки, внаслідок чого виникає потреба пошуку елементарних умовних співвідношень для побудови моделей поведінки користувачів, бо пошук таких залежностей виходить за межі цих алгоритмів.

Об'єкт дослідження – побудова моделі поведінки клієнтів.

Предмет дослідження – методи та алгоритми машинного навчання для роботи з великими обсягами даних і для пошуку прихованих залежностей у поведінці респондентів.

Мета роботи – розробити та перевірити групи методів для попереднього оброблення даних (кластеризація) та виявлення закономірностей планування зустрічей друзів на підставі аналізу їхнього щоденного руху.

Для досягнення зазначеної мети визначено такі *основні завдання дослідження:* створення та розуміння моделей поведінки користувачів; прогнозування майбутньої поведінки за допомогою створеного шаблону, що дасть змогу знаходити послідовності розташування місця потенційного клієнта у певний момент, упродовж дня чи наступного дня.

Наукова новизна отриманих результатів дослідження – запропоновано метод, орієнтований на створення та розуміння моделей поведінки користувачів, прогнозування їхньої майбутньої поведінки за допомогою створеного шаблону, що встановлює часові залежності, пов'язані з поведінкою клієнта, а саме визначає не тільки, що робить певний клієнт у певний момент, але й що він робить продовж дня та наступного дня.

Практична значущість результатів дослідження полягає в тому, що розроблений підхід виявляє закономірності планування зустрічей друзів на підставі аналізу їхнього щоденного руху, а з використанням алгоритму *k*-means можливо визначити, наскільки добре кожен об'єкт знаходиться у своєму кластері. Алгоритм вилучення асоціативних правил хоча і дає змогу знаходити статистично значущі правила та шукати тільки залежності, визначені загальним набором вхідних даних, однак має високу обчислювальну складність, якщо існує багато правил класифікації. Матриця невідповідностей показує відповідні результати прогнозування, а рівень помилок становить менше 6,7 %. Найбільша помилка трапляється для нульового класу.

Аналіз останніх досліджень та публікацій. Основними методами кластеризації є: ієрархічні, секціонування, штучні нейронні мережі, штучні нейронні мережі на підставі щільності або сітки [5].

Ієрархічна кластеризація створює ієрархію кластерів, або іншими словами, дерево кластерів, також зване дендрограмою. Кожен вузол кластера містить дочірні кластери; кластерні нащадки ділять вершини, що належать їх спільному предку. Цей підхід дає змогу досліджувати дані на різних рівнях деталізації. Недоліки пов'язані з тим, що більшість ієрархічних алгоритмів не повертаються до вже побудованих (проміжних) кластерів з метою їх вдосконалення; нечистотою критерію завершеності; проблемами масштабованості в разі застосування даних великого обсягу [2].

На відміну від ієрархічних методів, коли кластери будуються поступово, методи поділу кластеризації досліджують усі сегменти відразу [5]. Роблячи це, вони або намагаються ідентифікувати кластери шляхом ітеративного переміщення точок між підмножинами, або визначають кластери як області, щільно заповнені об'єктами. Алгоритми першого роду належать до кластеризації переміщення розділів. Водночас, вони поділяються на імовірнісний, *k*-means та *k*-medoid методи і концентруються на пристосуванні точок до відповідних скупчень, маючи тенденцію до побудови сегментів сферичної форми.

Алгоритми секціонування другого типу належать до групи методів секціонування на підставі щільності. Вони намагаються виявити щільно пов'язані компоненти даних, гнучкі з погляду їх форми. Методи засновані на групуванні сусідніх об'єктів у кластери на підставі їх локальної компактності, а не близькості [8]. Ці методи розглядають скупчення як ділянки густо розташованих об'єктів, які розділені на більш рідкісні, галасливі регіони. Основними перевагами методів кластеризації на підставі щільності є можливість виявлення кластерів вільної форми різного розміру та стійкості до шуму та викидів. До недоліків можна віднести високу чутливість до встановлення вхідних параметрів, поганий опис класів і непридатність для даних з великими розмірами.

Сіткові методи кластеризації працюють опосередковано, розділяючи простір елементів даних на кінцеву кількість комірок і залишаючи для подальшого оброблення ті комірки, які мають велику щільність об'єктів, а ізольовані елементи ігноруються. Просторовий розділ базується на характеристиках сітки, зібраних із вхідних даних. Методи кластеризації, засновані на сітці, мають такі переваги: нагромадження даних призводить до незалежності методу від їх порядку; розрахунок відстані не проводиться; можливість оброблення атрибутів різних типів; легко ідентифікувати сусідні скупчення [1]. Недоліки пов'язані з визначенням відповідного розміру конструкції решітки; виявлення скупчень з різною щільністю та формами; вибір умов об'єднання для формування ефективних кластерів.

Правила асоціації AR (англ. *Association Rules*) – це набір спеціальних правил, які дають змогу знаходити та описувати відповідність у великих наборах даних [13]. Основними поняттями в теорії асоціативних правил є предметний набір і транзакція. Тематичний набір – це непустий набір елементів, які можуть бути частиною транзакції:

$$I = \left\{ i_k, k = \overline{1,n} \right\}, \tag{1}$$

де: i_k – k-ий елемент, що входять до предметних наборів; n – кількість елементів набору I.

Транзакція є певним набором, який містить деякі елементи набору I, що відбуваються разом. Транзакція також має унікальний ідентифікатор TID (англ. *Transaction* ID).

У базі даних є певний набір транзакцій:

$$T = \left\{ t_i, i = \overline{1,m} \right\}, \tag{2}$$

де: t_i – i-ий набір транзакції; m – загальна кількість транзакцій.

Поняття множини та асоціативного правила тісно пов'язані з іншою її характеристикою – довірою, яку обчислюють як відношення множини, що має як умову, так і наслідок (іншими словами, це підтримка асоціативного правила), щоб підтримати множину, яка має тільки умову.

$$\begin{aligned} \mathrm{Conf}\left(\mathrm{X} \to \mathrm{Y}\right) &= \mathrm{Supp}\left(\mathrm{X} \to \mathrm{Y}\right) / \mathrm{Supp}\left(\mathrm{X}\right) = \\ &= \left|X(t) \cap Y(t)\right| / \left|X(t)\right|. \end{aligned} \tag{3}$$

Для визначення значущості правил використовуються порогові значення мінімальної підтримки та достовірності MinSupp та MinConf, які, зазвичай, визначаються користувачами системи або експертами, виходячи з власного досвіду:

$$Supp\left(X \to Y\right) \geq MinSupp; \tag{4}$$

$$Conf\left(X \to Y\right) \geq MinConf. \tag{5}$$

Методи пошуку асоціативних правил знаходять усі асоціації, які відповідають обмеженням підтримки та впевненості. Однак це призводить до потреби переглянути досить велику кількість асоціативних правил, які бажано скоротити так, щоб проаналізувати тільки найбільш значущі з них.

Серед основних алгоритмів генерування асоціативних правил виділяють AIS, SETM, Apriori, AprioriTid, AprioriHybrid [15]. Ефективність та доцільність використання кожного з них зумовлені структурою та обсягом набору даних, для яких здійснюється пошук асоціативних правил, оскільки основа цих методів знаходиться у різних принципах генерування та вибору предметних сукупностей – кандидатів.

Алгоритм AIS – це перший ефективний алгоритм пошуку асоціативних правил, робота якого складається з двох етапів:

- перший крок реалізує процедуру генерування частих предметних наборів,
- другий крок – побудова частих правил із заданою впевненістю.

Недоліком цього алгоритму є те, що в процесі пошуку правил він неодноразово проходить через один набір даних.

Алгоритм SETM, як і AIS, складається з двох етапів і виконує формування предметних наборів кандидатів на льоту, використовуючи мову інструмента SQL. У ньому зберігається копія тематичного набору кандидатів разом із TID у спеціальній, послідовній структурі. Після проходження всього набору даних проводиться підрахунок підтримки кандидатів шляхом сортування та агрегування отриманої структури. Недоліками алгоритму SETM, як і AIS, є багаторазові проходження через набір даних і генерація надлишкових кандидатів, які внаслідок не належать до частих предметних наборів.

Недоліки зазначених вище алгоритмів вирішуються алгоритмом Апріорі, запропонованим Р. Агравалем та Р. Срікантом. На відміну від AIS та SETM, він усуває генерування та підрахунок надмірної кількості кандидатів завдяки використанню антимонотонних властивостей та дає змогу значно зменшити множину частих наборів предметів і цим самим зменшити простір пошуку асоціативних правил. Властивість різноманітності стверджує, що якщо набір предметів Z не часто трапляється, то додавання якогось нового об'єкта Y до набору Z не змінює його частоту (відповідно, якщо Z не є частим набором предметів, то і ZY не є частим також). Модифікаціями класичного алгоритму Apriori є AprioriTid та AprioriHybrid.

За допомогою методу Апріорі реалізують пошук асоціативних правил. Оскільки розмір сучасних баз даних може досягати досить великих обсягів (гігабайти та терабайти), пошук асоціативних правил потребує ефективних алгоритмів, які є масштабованими і дають змогу знайти рішення цього завдання у прийнятний час. Одним із них є алгоритм Апріорі, який розробили для реляційних баз даних. Алгоритм дає змогу генерувати часті набори даних з таблиць транзакцій.

Алгоритм Апріорі використовує ітераційний підхід. На першому кроці алгоритму є одноелементні, часті набори даних, що позначаються набором L_1. На наступному кроці набір L_1 використовується для пошуку частих двоелементних наборів, з яких формується набір L_2, який, водночас, використовується для пошуку триелементних наборів L_3, і так далі, поки всі можливі часті k-елементи знайдено множини L_k.

Модель AOG – це орієнтований ациклічний граф, де кожна вершина графа відповідає змінній із заданими параметрами. У байєсівських мережах параметри подаються як локальний умовний розподіл ймовірностей значень змінних $P(Xi \mid F(Xi))$. А в гауссових мережах – як коефіцієнти лінійних рівнянь (для ребер) і дисперсії відхилень (для вершин). Побудова AOG-моделей відповідає проблемі відтворення моделі зі статистичних даних. Сюди входять методи відновлення моделі AOG "Collifinder" та "Proliferator-C", узагальнюючи метод Chow & Liu. Застосування Collifinder та Proliferator-C дає змогу розпізнавати транзитивні, синергетичні та комбіновані асоціації, а отже, забезпечує надійний та ефективний метод відтворення структур однопотокових моделей залежностей без тестів першого рівня.

Проблема описаних вище методів полягає у потребі задати елементарні умовні співвідношення для побудови графіка AOG-моделі. Пошук таких залежностей виходить за межі цих алгоритмів [6], [9], [10].

Результати дослідження та їх обговорення

Алгоритми розпізнавання шаблонів з навчанням припускають наявність історичної інформації, що дає змогу будувати статистичні моделі зв'язків $x \rightarrow y$, де $y \in Y$, Y спостерігаються дії користувача (відповіді) або моделюється випадкова величина $x \in X$, X – набір змінних (предиктори), за допомогою яких передбачається пояснити мінливість змінної y. Більшість моделей з викладачем розроблені так, що їх можна записати як

$$y = f(x, \beta) + \varepsilon, \qquad (6)$$

де: $f(x, \beta)$ – математична функція, вибрана з якогось довільного сімейства; β – вектор параметрів цієї функції; ε – помилки, які, зазвичай, породжують неупереджені, некорельовані випадкові процеси.

Під час побудови модель при фіксованих значеннях вибірки y мінімізують залишки деякої функції $Q(y, \beta)$. Унаслідок знайдено β. Це вектор з оптимальними оцінками параметрів моделі. Змінюючи форму функцій f і Q, можна отримати різні моделі, з яких перевага віддається найефективнішій моделі. Ця модель забезпечує неупереджені, точні та надійні прогнози відповіді y.

Метод пошуку залежностей. На підставі виконаного аналізу даних зібрано інформацію для заповнення розумного профілю користувача, а саме:

- розумні статуси – які місця відвідував користувач та можливість показувати їх у часовому рядку своїм друзям;
- де – тип бажаних місць для користувача;
- час – коли користувач найімовірніше доступний;
- використання даних для організації зустрічей на підставі розумного профілю користувача. Коли користувач домовляється про зустріч, програма може запропонувати: де зустрітися, виходячи із типу звичайної діяльності користувача;
- коли зустрічатися, відповідно до його вільного часу та найімовірніше вільного часу його друга;
- друзі, які підходять до виду діяльності, часу, місце перебування;
- інформація про зустріч.

Набір навчальних даних складається з шести параметрів:

- вхід – ім'я користувача (рядок),
- широта (подвійна),
- довгота (подвійна),
- PlaceType (рядок) – інформація з GoggleAPI,

- оцінка (ціле число) – інформація від GoggleAPI,
- часова позначка (час, тривалість).

Брали до уваги 292 зразки для навчання, що складається з денної маршрутизації для 30 користувачів. Для аналізу даних було запропоновано:

- групування даних за логіном;
- розділення чинників за логіном;
- розділення на чинники за місцем типу;
- розділення чинників за тарифами.

Для розпізнавання шаблонів використано триступеневий алгоритм:

- створення кластера користувачів – для пошуку подібної поведінки;
- побудова шаблону – для пошуку послідовності місця;
- наступний прогін PlaceType – для передбачення наступного стану користувача.

Створення кластера користувачів. Для кластеризації використано *R* та пакети factoextra та кластер. На рис. 1 показано результати ієрархічної кластеризації.

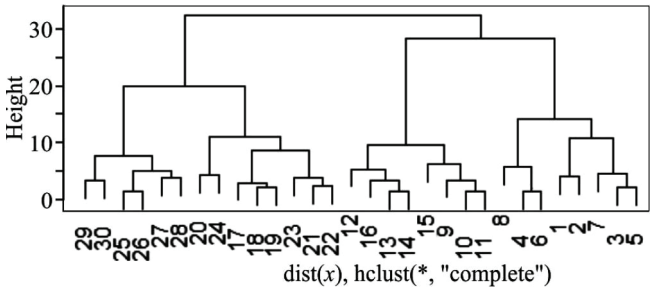

Рис. 1. Результат ієрархічної кластеризації

У роботі ми використовуємо *k*-means та *k*-medoid алгоритми розділення. По-перше, ми знайшли оптимальну кількість кластерів за допомогою методу Elbow та статистики розривів. Статистика розривів може бути застосована до будь-якого методу кластеризації. Він порівнює загальну варіацію внутрішнього кластера для різних значень *k* з їхніми очікуваними значеннями при нульовому еталонному розподілі даних (тобто розподіл без явної кластеризації). Довідковий набір даних генерується за допомогою моделювання методом Монте-Карло процесу відбору проб (рис. 2).

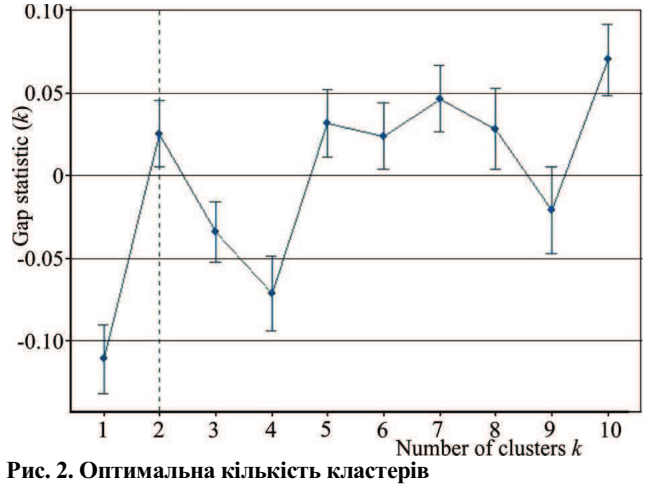

Рис. 2. Оптимальна кількість кластерів

Оптимальна кількість кластерів – 4. Далі пропонуємо аналіз *k*-means (рис. 3). Подано перетин кластерів 1 і 2. Візуалізація дає змогу наочно оцінити дані, наприклад, можна побачити, що дані утворюють кілька груп, тому можна зробити кластеризацію й навчити класифікатор на кожен кластер. Виходить проста двоступенева гібридна модель, за результатом кластеризації будується перший класифікатор на кілька класів (кластерів, які виявлені алгоритмом кластеризації), а потім для кожного класу/кластера використовують свій класифікатор для цільового поля. Як відомо [9], алгоритм *k*-means на вхід приймає кількість кластерів і розбиває дані на вказану кількість груп. Замість того, щоб перебирати кількість кластерів і шукати розбиття з найменшою вартістю, можна візуалізувати дані й побачити ці групи. Але є й інша проблема, наприклад, розбивши дані за відстанню Евкліда, можемо отримати дуже неявні межі кластерів, вони будуть дуже близько один до одного і тим самим точність системи загалом зменшується.

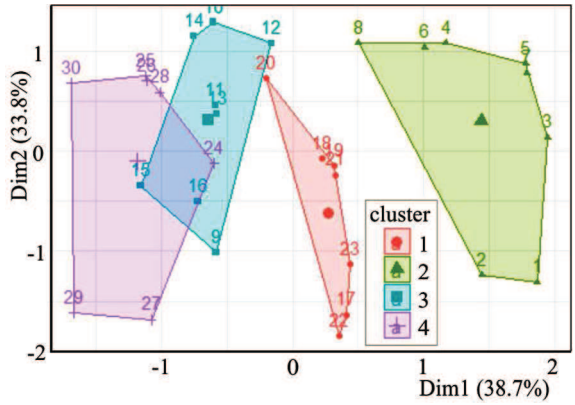

Рис. 3. Результати роботи алгоритму розділення *k*-means

Результат Medoids наведено на рис. 4. Для обчислення середньої ширини образу можна використовувати функцію образу в пакеті кластера. Підхід середнього образу вимірює якість кластеризації. Тобто він визначає, наскільки добре кожен об'єкт знаходиться у своєму кластері. Висока середня ширина образу вказує на якісну кластеризацію. Оптимальна кількість скупчень *k* – максимізує середній образ за діапазон можливих значень для *k*.

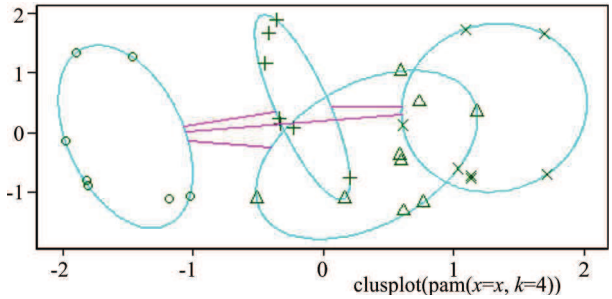

Рис. 4. Результати роботи алгоритму розділення *k*-medoid

Отже, результати алгоритмів розділення однакові. Спробуємо інші методи кластеризації.

Методи кластеризації на підставі моделі використовують ймовірнісну модель даних, виходячи з припущення, що вони мають певну кількість різних розподілів ймовірностей залежно від їх походження. Основою розбиття на кластери є пошук даних, що мають однаковий розподіл. Такі методи поділяють на статистичні методи та підходи, засновані на штучних нейронних мережах. В основі статистичних підходів знаходиться розрахунок імовірнісних параметрів під час визначення кластерів, тоді як нейронні мережі представлені як сукупність вхідних і вихідних нейронів із зваженими зв'язками. Перевагами методів, заснованих на штучних нейронних мережах, є здатність до адаптивного навчан-

ня, стійкість до шуму та викидів даних і паралельне оброблення інформації. Недоліки пов'язані зі складністю встановлення вхідних параметрів мережі та початкових ваг; залежність збіжності алгоритму від вхідних параметрів; здатність налаштованої та навченої мережі адаптуватися до нових вхідних даних; можливість оброблення тільки числових даних. Переваги статистичних методів містять лінійну складність алгоритмів; дуже хороші результати на реальних даних; забезпечення статистичної моделі даних і здатності обробляти пов'язані з ними невизначеності. Недоліки пов'язані зі складністю, тенденцією до зближення в місцевому оптимумі та необхідністю припустити нормальний розподіл вимірювань даних.

Результат роботи EM-алгоритму наведено на рис. 5.

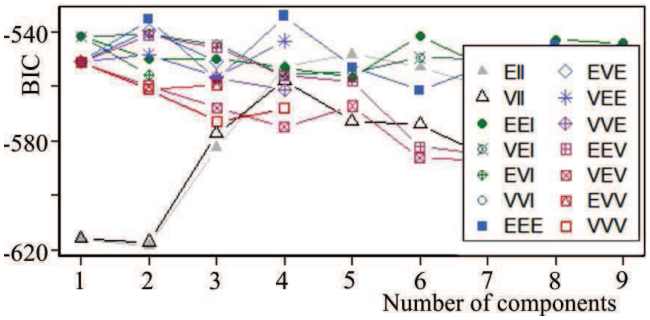

Рис. 5. Результати роботи EM-алгоритму

Нечіткий алгоритм *c*-середніх (Fuzzy *c*-means) – дає змогу отримати нечітку кластеризацію великих наборів числових даних, що дозволяє коректно визначати об'єкти на межах кластерів. Однак, виконання даного алгоритму вимагає серйозних обчислювальних ресурсів, а також початкового задавання кількості кластерів. Окрім цього, може виникнути неоднозначність з об'єктами, віддаленими від центрів всіх кластерів. В табл. 1 нечітку *c*-means кластеризацію з 4 кластерами.

Табл. 1. Кластерні центри

№	Index	Class	Login
1	2.963015	1.0800901	22.248811
2	18.828462	1.6122692	22.469047
3	25.866844	0.9582239	11.867454
4	11.346394	1.0694440	4.760382

Найближче сильне скупчення [11]: 1 1 1 1 1 1 1 1 4 4 4 4 4 4 4 2 2 2 2 2 2 3 3 3 3 3 3 3.

Аналіз показує, що основною проблемою всіх алгоритмів кластеризації є їх масштабованість зі збільшенням обсягу оброблених даних. Окрім цього, основними проблемами більшості з них є складність налаштування оптимальних вхідних параметрів (для алгоритмів щільності, сітки чи моделі), ідентифікація кластерів різної форми та щільності (алгоритми розподілу, алгоритми на підставі сітки), нечіткі критерії завершення (ієрархічний, розділовий та на підставі моделі). Оскільки процедура кластеризації є тільки одним із етапів оброблення даних системи загалом, обраний алгоритм повинен бути простим у використанні та простим для налаштування вхідних параметрів. Проведені нами дослідження показують, що ієрархічні методи кластеризації даних містять ряд алгоритмів, придатних як для оброблення даних невеликого обсягу, так і для аналізу великих даних, що є актуальним у галузі соціальних мереж.

Побудова шаблону. Навчання асоціативних правил – метод машинного навчання, що ґрунтується на правилах для знаходження цікавих відношень між змінними у великих базах даних. Метою є ідентифікація сильних правил, які виявляються в базах даних з використанням деяких вимірів зацікавленості. Однак, існують об'єктивні та суб'єктивні міри відповідності асоціативного правила. Завданнями є зазначена вище підтримка та впевненість. Суб'єктивними мірками значущості є ліфт і важелі. Підйом визначають відношенням збереження асоціативного правила до стану підтримки продукту та ефекту окремо:

$$Lift(X \to Y) = \frac{Supp(X \to Y)}{Supp(X) \cdot Supp(Y)}. \quad (7)$$

Піднесення ($Lift$) – це так звана узагальнена міра зв'язку між двома предметними сукупностями. Його значення можна інтерпретувати так:

якщо $Lift(X \to Y) = 1$, то $Supp(X \to Y) = Supp(X) \cdot Supp(Y)$, (8)

тобто стан і наслідок не залежать один від одного;

якщо $Lift(X \to Y) > 1$, то $Supp(X \to Y) > Supp(X) \cdot Supp(Y)$, (9)

тобто наслідок позитивно залежить від стану;

якщо $Lift(X \to Y) < 1$, то $Supp(X \to Y) < Supp(X) \cdot Supp(Y)$, (10)

тобто наслідок негативно залежить від стану.

Результати виконаного аналізу даних наведено нижче (рис. 6 та табл. 2).

Набори предметів (LHS)	Набори предметів як наслідок (RHS)
[1] "{Закупки}" "{Водіння}" "{Розваги}" "{Спорт}"	[1] "{Фінанси}" "{Спорт}" "{Розваги}" "{Водіння}"
[5] "{Фінанси}"	[5] "{Покупки}"

Рис. 6. Структура асоціативних правил

Табл. 2. Параметри асоціативних правил

LHS	RHS	підтримка	довіра	піднесення	номер
[1]	{Охорона здоров'я} => {Фінанси}	0.007692308	0.09090909	0.5371901	1
[2]	{Фінанси} => {Охорона здоров'я}	0.007692308	0.04545455	0.5371901	1
[3]	{Спорт} => {Розваги}	0.015384615	0.14285714	1.1607143	2
[4]	{Розваги} => {Спорт}	0.01538461	0.12500000	1.1607143	2
[5]	{Спорт} => {Фінанси}	0.015384615	0.14285714	0.8441558	2
[6]	{Фінанси} => {Спорт}	0.015384615	0.09090909	0.8441558	2
[7]	{Забави} => {Фінанси}	0.007692308	0.06250000	0.3693182	1
[8]	{Фінанси} => {Розваги}	0.007692308	0.04545455	0.3693182	1
[9]	{Покупки} => {Водіння}	0.046153846	0.46153846	1.8181818	6
[10]	{Водіння} => {Покупки}	0.046153846	0.18181818	1.8181818	6
[11]	{Відпочинок} => {Фінанси}	0.015384615	0.11111111	0.6565657	2
[12]	{Фінанси} => {Відпочинок}	0.015384615	0.09090909	0.6565657	2

Отже, правила 3, 4, 9, 10 важливі для аналізу. Далі знайдено підтримку наступних правил (рис. 7). Окрім згаданого вище прикладу з аналізу ринкових кошиків, асоціативні правила застосовуються сьогодні у багатьох областях, в т.ч. й Web Mining, виявлення вторгнень, безперервне виробництво та біоінформатику. На відміну від пошуку шаблону послідовності, виведення послідовності, навчання асоціативних правил, як правило, не враховує порядок елементів у межах транзакції чи транзакцій.

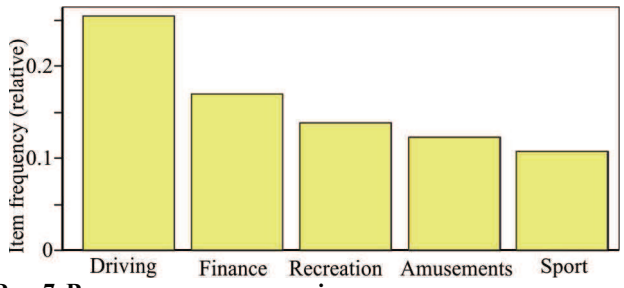

Рис. 7. Важливе значення асоціативних правил

В іншому випадку значення правил є важелем, який запропонував Г. Пятецький-Шапіро [6]. Кредитне плече – це різниця між частотою, з якою умова і наслідок з'являються разом, тобто підтримкою асоціативного правила, і добутком підтримки умови та ефекту окремо:

$$Lev(X \to Y) = Supp(X \to Y) - Supp(X) \cdot Supp(Y). \quad (11)$$

Поліпшення – це відношення частоти спостережуваного виконання правила до продукту виникнення стану та ефекту окремо.

$$I(A \to B) = \frac{S(A \to B)}{P(A) \cdot P(B)}. \quad (12)$$

Власний алгоритм. Запропоновано алгоритм вилучення асоціативних правил. Для взаємозв'язку зі схемою $R = \{A_i, dom(A_i)\}, i = \overline{1,m},$ дає змогу знаходити статистично значущі правила, що відображають залежність атрибута A_m на атрибути $A_1, A_2, \ldots, A_{m-1}$, тобто залежність виду $A_1, A_2, \ldots, A_{m-1} \to A_m$. Як міру статистичної значущості використовують інформаційний показник Кулбаха-Лейблера. Алгоритм дає змогу шукати тільки залежності, визначені загальним набором вхідних даних; окрім цього, він має високу обчислювальну складність, якщо існує багато правил класифікації.

Для величезного набору даних з невідомою структурою асоціації з високою підтримкою окремих подій практично відсутні. Отже, такі асоціації, хоча вони можуть мати інтерес, будуть виключені з розгляду, оскільки вони не будуть відповідати певному мінімальному порогу підтримки S_{min}.

Для вирішення цієї проблеми ми пропонуємо знайти асоціативні правила не тільки для окремих предметів, а й для їх ієрархії. Якщо на нижчих ієрархічних рівнях немає таких цікавих асоціацій, то вони можуть виникати на вищих рівнях. Іншими словами, підтримка окремого об'єкта завжди буде меншою, ніж підтримка групи, до якої він належить:

$$S(I) > S(i_j), \quad (14)$$

де: I – група знаходиться в ієрархії; i_j – j-ий елемент, внесений до даної групи. Причини цього очевидні: загальна підтримка групи дорівнює сумі підтримки для внесених до неї предметів:

$$S(I) = \sum_{j=1}^{n} i_j, \quad (15)$$

де n – кількість елементів у групі. Асоціативні правила, знайдені для об'єктів чи подій, розташованих на різних ієрархічних рівнях, називають багаторівневими правилами. Спускаючись до нижчих рівнів абстракції, аналізуються нащадки тільки тих категорій та підкатегорій, які є частими наборами, тобто існує принаймні заздалегідь визначена кількість разів, де k – кількість рівнів.

Як на недолік ієрархічного підходу до пошуку асоціативних правил іноді вказують на те, що отримані правила здебільшого стосуються не окремих предметів, а їхніх груп, а це не завжди відповідає вимогам аналізу даних. Однак у деяких предметних областях кількість окремих найменувань може бути настільки значною, що виявлення асоціативних правил навіть на високому рівні узагальнення – це вже успіх. Окрім цього, технологія бізнес-аналізу більшою мірою орієнтована на узагальнені дані, тому виробники й тим більше конкретні моделі рідко цікавлять дослідника при побудові відповідних асоціацій.

Існує декілька підходів до пошуку ієрархічних асоціативних правил. Повторювані методи часто використовують, коли набори предметних предметів досліджують на кожному ієрархічному рівні, від першого до рівня з найбільшою деталізацією. Простіше кажучи, як тільки виявляються всі часті набори предметів на першому рівні, починається пошук популярних наборів предметів на другому тощо. На кожному рівні для пошуку частих наборів можна використовувати будь-який алгоритм, такий як Apriori та його модифікації. Відомо декілька стратегій проведення правил пошуку.

1. Використовуйте той самий мінімальний поріг для підтримки $S^k_{\min} = const$ на всіх ієрархічних рівнях. Під час пошуку правил встановлюється один раз певний мінімальний поріг підтримки (наприклад, 5 %), коли досягнення якогось набору вважається частим, воно не входить до списку правил. Перевага підходу: висока швидкість аналізу предметної області зумовлена відмовою від оцінки часткових наборів, отриманих з тих, які недостатньо поширені. Відсутність: ризик передачі тонких асоціацій на нижчих рівнях ієрархії, а саме перевага користувачів одного виробника або підключення певних моделей товарів. Іноді цю ваду намагаються обійти, зменшивши мінімальний рівень підтримки. Як результат, поява десятків і сотень вільних правил з низькою підтримкою та ймовірністю.
2. Зниження порогу мінімальної підтримки при переході на нижчі рівні ієрархії. Він може бути реалізований індивідуально для кожної підгрупи. Функціональний тип порогового зменшення, як правило, пов'язаний з кількістю підкатегорій або серійним номером рівня. Іншим варіантом функціонального підходу є встановлення порогових значень для мінімальної підтримки виключно залежно від рівня, незалежно від того, яка б підкатегорія не була $S^k_{min} = S^1_{min}/k$. Перевага такого підходу: він дає змогу пройти набагато далі в ієрархії у пошуках асоціацій. Однак у разі індивідуального завдання граничного рівня для кожної підкатегорії ця процедура відніме левову частку часу. У разі встановлення порогу, залежно від рівня або кількості підкатегорій, не враховуються індивідуальні переваги та недоліки окремих виробників і моделей.

3. Міжшарова фільтрація заснована на так званому проході рівня (рівень проходу). Відносно високий для верхніх рівнів ієрархії, граничний рівень підтримки залишається внизу під час першого проходження. Потім при кожному проходженні цей рівень знижується. Ці низькопрофільні предмети, які мають асоціації, в такий спосіб ідентифікуються раніше, ніж їхні батьківські категорії, які можуть не мати необхідної підтримки. Для підключення рівнів довіри представників різних рівнів рекомендується здійснити три-чотири проходження бази даних транзакцій.

Пошук PlaceType передбачення. Створений шаблон не відповідає на питання про те, як пов'язати асоціацію з клієнтом, а також встановлює часові залежності, тобто – не лише для того, щоб відповісти на питання *Що* робить певний клієнт у певний момент, але *Що* він робить впродовж дня. А якщо є можливість – передбачити, що він робитиме у *Наступний День*.

Відповіді на це питання забезпечують використання послідовних шаблонів, заснованих на теорії асоціацій, обов'язковими полями яких є дата/час та ідентифікатор користувача. При розгляді послідовностей транзакцій використовується одне припущення – один і той самий клієнт не виконує дві різні транзакції одночасно.

Послідовність S називають максимальною, якщо вона не міститься в жодній іншій послідовності. Послідовність S називають клієнтом, якщо вона, крім набору об'єктів, дати та часу, містить також ідентифікатор користувача.

Послідовність S_1 міститься в послідовності S_2, якщо всі набори предметів S_1 містяться в наборах S_2 предметів. Послідовність S_1 є послідовною, якщо всі набори предметів містяться в наборах предметів.

Наприклад, послідовність < (3); (4, 5); (8) > міститься в послідовності < (7); (3, 8); (9); (4, 5, 6); (8) >, оскільки (3) (3, 8), (4, 5) (4, 5, 6) та (8) (8).

Однак < (3); (5) > < (3, 5) > і навпаки, оскільки в першій послідовності предмети 3 і 5 купувались один за одним, а в другій – разом.

Послідовність S називають підтримуваною, якщо вона міститься в її клієнтській послідовності. Потім підтримка послідовності визначається як кількість клієнтів, які її підтримують і, зазвичай, виражається у відсотках від загальної кількості клієнтів. Отже, концепція підтримки послідовних шаблонів дещо відрізняється від аналогічного поняття про асоціативні правила.

Для бази даних користувальницьких транзакцій завдання пошуку послідовних шаблонів полягає у визначенні максимальної кількості послідовностей серед усіх, що мають підтримку заданого вище порогу. Кожна така максимальна послідовність і буде послідовним візерунком. Далі ми будемо називати послідовності, які задовольняють обмеження мінімальної підтримки, часто (за аналогією з частим траплянням множин у теорії асоціативних правил).

Процес пошуку послідовних шаблонів складається з таких етапів:

1. *Сортування*. Транзакції вихідної бази даних сортуються за кодом користувача, а транзакції кожного користувача – за датою та часом. Результат – база даних послідовностей клієнтів.
2. *Пошук частих предметних наборів F*. Часто називають предметними наборами, які були придбані за кількістю клієнтів, що перевищує мінімально допустиме значення. Вибраний набір частих предметних наборів перекладається у числове або символьне подання.
3. *Трансформація*. Необхідно визначити, які з найчастіших послідовностей містяться в послідовності клієнта. Для цього кожна транзакція в клієнтській послідовності замінюється безліччю її частих наборів предметів. Якщо в транзакції немає частотного набору, він більше не розглядається. Більше того, якщо конкретний клієнт у послідовності не має єдиного набору частот, він також виключається з розгляду. Після перетворення кожна послідовність клієнтів є упорядкованим набором частих наборів.
4. *Пошук частих послідовностей*. Часті послідовності шукаються на безлічі частих наборів предметів. Мінімальна частота – параметр алгоритму.
5. *Пошук максимуму послідовностей*. Серед частих послідовностей є максимум. Іноді цей етап поєднують з попереднім, щоб скоротити час, витрачений на обчислення не максимальних послідовностей.

Найбільш проблематичним кроком у пошуку послідовних шаблонів є ідентифікація частих послідовностей, оскільки велика кількість предметних наборів вимагає розгляду величезної кількості можливих комбінацій та декількох проходів через набір транзакцій. Кожен уривок починається з початкового набору послідовностей, які використовуються для генерування нових потенційних частих послідовностей, які називають послідовностями кандидатів або просто кандидатами. Для цього вони обчислюють свою підтримку і, після завершення проходження, визначають, чи є часто виявлені кандидати часто. Виявлені часті послідовності кандидатів будуть відправною точкою для нового проходу.

Для перевіреного набору даних після кроку 3, беручи до уваги виключення користувача, були отримані послідовності клієнта (табл. 3).

Табл. 3. Послідовність клієнта

Користувач	Послідовність
1	< {1,5}; {2}; {3}; {4} >
3	< {1}; {3}; {4}; {3, 5} >
5	< {1}; {2}; {3}; {4} >
6	< {1}; {3}; {5}; {4} >
8	< {4}; {5} >

Пошук частих послідовностей відбувається від рівня 1 до максимально можливого. Результати послідовних передач наведено в табл. 4. Наведено підтримку Sup. кожного правила.

Табл. 4. Матриця прогнозування

1-послідовність		2-послідовність		3-послідовність		4-послідовність	
F_1	Sup.	F_2	Sup.	F_3	Sup.	F_4	Sup.
1	4	1; 2	2	1;2;3	2	1;2;3;4	2
2	2	1; 3	4	1; 2;4	2		
3	4	1; 4	3	1; 3;4	3		
4	4	1; 5	3	1; 3;5	2		
5	4	2; 3	2	2; 3;4	2		
		2; 4	2				
		3; 4	3				
		3; 5	2				
		4; 5	2				

Отже, максимум послідовностей <1; 2; 3; 4>, <1; 3; 5> і <4; 5>, оскільки вони не містяться в послідовностях більшої довжини. Їх будуть шукати послідовні шаблони.

Обговорення результатів дослідження. Дослідження проводили на даних, що збиралися з мобільного додатка – Sponter, який доступний для iOS та Android та використовує технологію визначення місцезнаходження GPS, щоб повідомляти про місцезнаходження в режимі реального часу тих, хто прийняв ваше запрошення приєднатися до відповідного кола та поділитися своїм місцезнаходженням. Основними особливостями програми є обмін даними про місцезнаходження та організація зустрічей. Користувачі можуть миттєво відкрити програму та побачити, де перебувають інші учасники. Користувачі можуть вибрати чи не ділитися своїм місцезнаходженням з якимись конкретними учасниками в будь-який конкретний час. Додаток дає змогу користувачам створювати геозони, які попереджають їх, коли інший входить або залишає інше місце. Окрім цього, мобільний додаток має розумні статуси, які визначають активність користувача на підставі місць, які вони відвідують.

Sponter збирає статистику розташування користувачів за умов:

- коли додаток активний, ми відстежуємо зміну місцезнаходження та записуємо це в базу даних;
- коли додаток закрито, ми зберігаємо в базі даних значні зміни місцезнаходження;
- коли програма перебуває в автономному режимі, ця інформація згодом оновлюється та оновлюється в базі даних;
- коли користувач перетинає зони геозаборів, програма прокидається і надсилає фактичне положення до бази даних;
- завдяки розташуванню програма збирає дані з давача активності.

Процес розпізнавання шаблонів складається з трьох етапів – створення кластера користувачів, часте створення шаблонів, наступний прогін PlaceType. При побудові моделі потрібно перевіряти точність. Тому ми ділимо наші дані на дві частини: навчання (80 %) та тестування (20 %).

Під час оцінювання параметра y модель розраховує ймовірність зчитування букви, а не конкретне значення 0 або 1. Потрібно визначити поріг ймовірності, далі користувача можна віднести до групи 0 або 1. Тепер, як порогове значення, пороговий параметр становить 0,09. Модель розпізнавання виглядає так:

якщо $y \leq$ поріг, тоді відповідь = 0,

якщо $y >$ поріг, то відповідь = 1;

Порівняємо результати прогнозної моделі з реальними даними. Непередбачені ситуації фактичних і прогнозованих значень відповіді наведено в табл. 5.

Табл. 5. Матриця невідповідностей

Факт/Прогноз	0	1
0	18	2
1	2	37

Отже, помилка прогнозування не є великою. Але ми можемо передбачити тільки PlaceType, а не наявне місце користувача впродовж наступного періоду часу.

Порівняємо наші результати з відомими методами. У роботах [7], [14] використовують двоступеневу модель для розпізнавання образів людини. Перша частина моделі, заснована на розширенні ConvNets до 3D-випадку, автоматично вивчає просторово-часові особливості. Потім другий крок полягає у використанні цих вивчених особливостей для навчання рекурентної моделі нейронної мережі для класифікації всієї послідовності. Виступи оцінюються за набором даних. Матриця невідповідностей для цього набору даних з використанням двоступеневого алгоритму, наведеного в роботі [14], показує середню точність 67,9 та 78,2 для запропонованого алгоритму.

Висновки

Продемонстровано результати розпізнавання шаблонів поведінки. Для вирішення проблеми використано ансамбль моделей. Зокрема, для вирішення поставлених задач використано кластеризацію, асоціативні правила та послідовні правила, алгоритм k-means для групування користувачів.

Запропоновано послідовні асоціативні правила для кожного кластера окремо. Введено поняття правил асоціації; розроблено метод пошуку залежностей для визначення пріоритетних даних, що впливатимуть на прогнозоване місце користувача. Досліджено, що матриця невідповідностей демонструє відповідні результати прогнозування, рівень помилок менше 6,7 %. Доведено, що найбільша помилка трапляється для нульового класу.

Заплановано наступні дослідження, орієнтовані на прогнозування *Place*, що пов'язане з інформацією про передбачуваний *PlaceType*.

References

[1] Bonchi, F., Castillo, C., Gionis, A., & Jaimes, A. (2011). Social Network Analysis and Mining for Business Applications. *ACM Transactions on Intelligent Systems and Technology, 1*(3), 1–37. https://doi.org/10.1145/1961189.1961194

[2] Hardiman, S. J., & Katzir, L. (2013). Estimating clustering coefficients and size of social networks via random walk. *Proceedings of the 22nd International Conference on World Wide Web* (WWW'2013), 539–550. https://doi.org/10.1145/2488388.2488436

[3] Hrytsiuk, Yu. I., & Grytsyuk, P. Yu. (2019). The methods of the specified points of the estimates of the parameter of probability distribution of the random variable based on a limited amount of data. *Scientific Bulletin of UNFU*, 29(2), 141–149. https://doi.org/10.15421/40290229

[4] ISO/IEC TR 24028:2020. Information technology – Artificial intelligence – Overview of trustworthiness in artificial intelligence. *International Organization for Standardization and International Electrotechnical Commissio* (англ.). May 2020. Retrieved from: https://www.iso.org/obp/ui/#iso:std:77608:en

[5] Jadhav, B. S., Bhosale, D. S., & Jadhav, D. S. (2016). Pattern based topic model for data mining. *International Conference on Inventive Computation Technologies* (ICICT'2016), 1–6. https://doi.org/10.1109/inventive.2016.7824855

[6] Maulik, U., & Bandyopadhyay, S. (2000). Genetic algorithm-based clustering technique. *Pattern Recognition, 33*(9), 1455–1465. https://doi.org/10.1016/s0031-3203(99)00137-5

[7] Melnykova, N., Marikutsa, U., & Kryvenchuk, U. (2018). The New Approaches of Heterogeneous Data Consolidation. *IEEE 13th International Scientific and Technical Conference on Computer Sciences and Information Technologies* (CSIT'2018), 408–411. https://doi.org/10.1109/stc-csit.2018.8526677

[8] Newman, M. E. J. (2003). Mixing patterns in networks. *Physical Review E, 67*(2), 113–126. https://doi.org/10.1103/physreve.67.026126

[9] Osman, Ahmed M. Shahat. (2019). A Novel Big Data Analytics Framework for Smart Cities. *Future Generation Computer Systems, 91*, 620–33. https://doi.org/10.1016/j.future.2018.06.046

[10] Ramírez-Rubio, R., Aldape-Pérez, M., Yáñez-Márquez, C., López-Yáñez, I., & Camacho-Nieto, O. (2017). Pattern classification using smallest normalized difference associative memory. *Pattern Recognition Letters, 93*, 104–112. https://doi.org/10.1016/j.patrec.2017.02.013

[11] Ranjith, K. S., Zhenning, Y., Caytiles, R. D., & Iyengar, N. C. S. N. (2017). Comparative Analysis of Association Rule Mining Algorithms for the Distributed Data. *International Journal of Advanced Science and Technology, 102*, 49–60. https://doi.org/10.14257/ijast.2017.102.05

[12] Shakhovska, N., Fedushko, S., Greguš ml., M., Melnykova, N., Shvorob, I., & Syerov, Y. (2019). Big Data analysis in development of personalized medical system. *Procedia Computer Science, 160*, 229–234. https://doi.org/10.1016/j.procs.2019.09.461

[13] Shakhovska, N., Kaminskyy, R., Zasoba, E., & Tsiutsiura, M. (2018). Association Rules Mining in Big Data. *International Journal of Computing, 17*, 25–32.

[14] Yang, T., Hou, Z., Liang, J., Gu, Y., & Chao, X. (2020). Depth Sequential Information Entropy Maps and Multi-Label Subspace Learning for Human Action Recognition. *IEEE Access, 8*, 135118–135130. https://doi.org/10.1109/access.2020.3006067

[15] Yang, X., Lin, X., & Lin, X. (2019). Application of Apriori and FP-growth algorithms in soft examination data analysis. *Journal of Intelligent & Fuzzy Systems, 37*(1), 425–432. https://doi.org/10.3233/jifs-179097

N. B. Shakhovska, N. I. Melnykova

Lviv Polytechnic National University, Lviv, Ukraine

METHODS OF BUILDING A MODEL OF USER BEHAVIOR

The number of clustering methods and algorithms were analysed and the peculiarities of their application were singled out. The main advantages of density based clustering methods are the ability to detect free-form clusters of different sizes and resistance to noise and emissions, and the disadvantages include high sensitivity to input parameters, poor class description and unsuitability for large data. The analysis showed that the main problem of all clustering algorithms is their scalability with increasing amount of processed data. The main problems of most of them are the difficulty of setting the optimal input parameters (for density, grid or model algorithms), identification of clusters of different shapes and densities (distribution algorithms, grid-based algorithms), fuzzy completion criteria (hierarchical, partition and model-based). Since the clustering procedure is only one of the stages of data processing of the system as a whole, the chosen algorithm should be easy to use and easy to configure the input parameters. Results of researches show that hierarchical clustering methods include a number of algorithms suitable for both small-scale data processing and large-scale data analysis, which is relevant in the field of social networks. Based on the data analysis, information was collected within fill a smart user profile. Much attention is paid to the study of associative rules, based on which an algorithm for extracting associative rules is proposed, which allows to find statistically significant rules and to look only for dependencies defined by a common set of input data, and has high computational complexity if there are many classification rules. An approach has been developed that focuses on creating and understanding models of user behaviour, predicting future behaviour using the created template. Methods of modelling pre-processing of data (clustering) are investigated and regularities of planning of meetings of friends on the basis of the analysis of daily movement of people and their friends are revealed. Methods of creating and understanding models of user behaviour were presented. The *k*-means algorithm was used to group users to determine how well each object lay in its own cluster. The concept of association rules was introduced; the method of search of dependences is developed. The accuracy of the model was evaluated.

Keywords: pattern sampling, sequential associative analysis, clustering.

Інформація про авторів:

Шаховська Наталя Богданівна, д-р техн. наук, професор, завідувач кафедри систем штучного інтелекту.
Email: nataliya.b.shakhovska@lpnu.ua; https://orcid.org/0000-0002-6875-8534

Мельникова Наталія Іванівна, канд. техн. наук, доцент, кафедра систем штучного інтелекту.
Email: nataliia.b.melnykova@lpnu.ua; https://orcid.org/0000-0002-2114-3436

Цитування за ДСТУ: Шаховська Н. Б., Мельникова Н. І. Методи побудови моделі поведінки користувачів. *Український журнал інформаційних технологій*. 2020, т. 2, № 1. С. 43–51.

Citation APA: Shakhovska, N. B., & Melnykova, N. I. (2020). Methods of building a model of user behavior. *Ukrainian Journal of Information Technology, 2*(1), 43–51. https://doi.org/10.23939/ujit2020.02.043