

ДОДАТОК А

СЛАЙДИ ПРЕЗЕНТАЦІЇ

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Атестаційна робота магістра

Аналіз алгоритмів пошуку в структурі даних для створення
зручного механізму опитування користувачів мереж

Науковий Керівник:
доцент

Ревенчук І.А.

Виконав:
студент групи ІПЗм-18-2

Абраменко Р.О.

2020

Актуальність дослідження

- Необхідність синхронізувати і організувати на єдиній критеріальною основі вибір методу кластеризації на підставі даних аналізованої вибірки;
- необхідність забезпечення користувача якісним рішенням завдання аналізу для опитувань;
- необхідність створення універсального методу кластеризації. Окремі методи добре працюють на певних даних, але не є універсальними;
- необхідність аналізу складних наборів даних, з пересічними класами.

Мета роботи

Аналіз алгоритмів пошуку в структурі даних для створення зручного механізму опитування користувачів мереж та побудова програмної системи за результатами аналізу.

3

Постановка задачі

- Аналіз та розробка модифікованого алгоритму до кластеризації даних на основі алгоритму Хамелеон;
- формування вимог до програмної системи;
- опис програмної реалізації;
- спроектувати та реалізувати програмну систему опитувань користувачів мереж.

4

Кластерний аналіз

Кластерний аналіз - це завдання розділити заданий зразок об'єктів на підмножини, що називаються кластерами, так, що кожен кластер складається з подібних об'єктів, а об'єкти різних кластерів значно відрізняються.

Спектр використання кластерного аналізу дуже великий і ця багатофункціональність використання призвела до появи значної кількості розрізнених визначень, методів і підходів, які ускладнюють єдине застосування послідовного тлумачення кластерного аналізу.

Завдання кластерного аналізу:

- Розробка типології чи класифікації;
- вивчення корисних концептуальних схем для групування об'єктів;
- генерація гіпотез на основі дослідження даних;
- тестування гіпотез або досліджень.

5

Етапи кластерного аналізу

- Підбір колекцій з метою кластеризації;
- встановлення великої кількості змінних, відповідно до яких об'єкти у вибірці будуть оцінюватися;
- обчислення значень того чи іншого рівня подібності між об'єктами;
- використання методу кластерного аналізу з метою формування груп подібних об'єктів;
- контроль достовірності результатів.

6

Вимоги кластерного аналізу до даних

- Показники не потрібно співвідносити між собою;
- показники повинні бути безрозмірними;
- розподіл показників має бути близьким до нормального;
- показники необхідні для задоволення умов стійкості, що означає відсутність впливу на їх значення несподіваних факторів;
- вибірка має бути однорідна.

7

Основні підходи кластерного аналізу

- Ймовірнісний підхід;
- підхід, в якому використовується аналогія з центром ваги;
- підхід, заснований на теорії графів;
- ієрархічний підхід;
- підхід, заснований на понятті найближчого сусіда;
- нечіткі алгоритми кластерного аналізу;
- підхід з використанням штучних нейронних мереж;
- еволюційний підхід.

8

Вимоги до алгоритмів кластеризації

- Масштабованість;
- можливість роботи з характеристиками різних типів;
- розпізнавання кластерів довільної форми;
- мінімальні вимоги до області дослідження для визначення вхідних параметрів;
- нечутливість до порядку записів, що входять;
- висока розмірність;
- простота використання та інтерпретації.

9

Алгоритм Хамелеон

Хамелеон – це новий ієрархічний алгоритм, який долає обмеження існуючих алгоритмів кластеризації.

Основним фактором методу Хамелеона вважається той факт, що він забезпечує одночасно взаємозв'язок і близькість при визначенні подібних пар кластерів.

Хамелеон знаходить кластери двофазним алгоритмом. На першому кроці Хамелеон використовує алгоритм розбиття графів для кластеризації великих чисел на досить малі підкласи. На другому кроці алгоритм використовується для пошуку природних кластерів за допомогою послідовного поєднання придбаних малих підкласів.

10

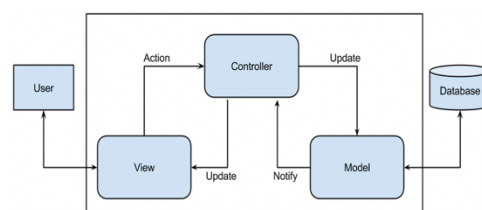
Етапи алгоритму Хамелеон

- побудова графа;
- огрубіння графа;
- початковий поділ графа;
- відновлення графа та вдосконалення поділу графа;
- об'єднання подібних класів для отримання остаточного розбиття.

11

Архітектура ПЗ

- Програмна система побудована на патерні MVC;
- серверна частина обробляє дані та надає клієнту;
- база даних містить усю інформацію;
- Веб-застосунок виконує роль представлення даних користувачеві.



12

Фрагмент коду алгоритму Хамелеон

```
run() {
  this.initCompleteGraph();
  this.runKnn();
  this.initClusters();

  while (this.clusters.length < this.initNrOfClusters) {
    const clusterToPartition = this.clusters.map()
      .max(Comparator.comparing((c) => c.getPoints().size()));
    const twoClusters = partitionCluster(clusterToPartition);
    this.clusters.push(twoClusters);
    this.clusters.pop();
  }

  while (this.clusters.length > this.resultNrOfClusters) {
    const twoClusters = findTwoClustersToConnect();
    const resultCluster = mergeTwoClusters(twoClusters[0], twoClusters[1]);
    this.clusters.add(resultCluster);
    this.clusters.removeAll(twoClusters);
  }

  return this.clusters;
}
```

13

Інтерфейс системи Survey

The screenshot displays the 'Survey Designer' interface. At the top, there's a navigation bar with tabs: 'Survey Designer' (active), 'Test Survey', 'JSON Editor', and 'Embed Survey'. Below this is a toolbar with buttons: 'Undo', 'Redo', 'Survey Settings', and 'Save Survey'. The main workspace shows a 'welcome screen' with a title 'This is travel quiz' and a subtitle 'Enter a description'. Below the title are three questions:

1. Full Name *
2. How old are you? *
3. Do you travel alone? *

On the right side, there's a 'TOOLBOX' panel with a list of input types: Single Input, Checkbox, Radiogroup, Dropdown, Comment, Rating, Image picker, Boolean, Image, HTML, Expression (read-only), File, Matrix (single choice), Matrix (multiple choice), and Matrix (dynamic rows). Each item has a small icon next to it.

Рисунок 1 – Головний екран системи побудови опитування

14

Інтерфейс системи Survey

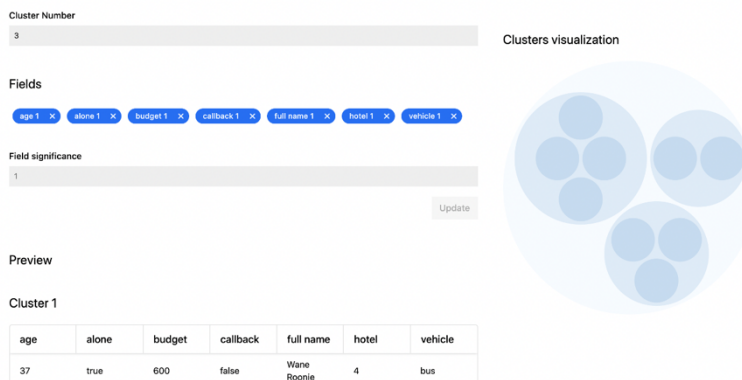


Рисунок 2 – Сторінка аналізу даних

15

Висновки

В результаті проведеної роботи було проведено аналіз алгоритмів пошуку в структурі даних для створення зручного механізму опитування користувачів мереж та побудова програмної системи за результатами аналізу. Було приведено вирішення завдання кластеризації великих обсягів даних, а також вибірок з різними статистичними характеристиками.

Були опубліковані тези доповіді на молодіжному форумі «Радіoeлектроніка та молодь у ХХІ ст» та на міжнародній студентській науковій конференції «Діджиталізація науки як виклик сьогодення».

16

ДОДАТОК Б

ЛІСТИНГ КОДУ

Нижче наведено програмний код для кластеризації даних.

```
class Chameleon {
  constructor(k, initNrOfClusters, resultNrOfClusters, points) {
    this.k = k;
    this.initNrOfClusters = initNrOfClusters;
    this.resultNrOfClusters = resultNrOfClusters;
    this.points = points;
    this.graph = [points.size()][points.size()];
    this.knnGraph = [points.size()][points.size()];
    this.clusters = [];
  }

  initCompleteGraph() {
    for (let i = 0; i < this.points.length; i++) {
      for (let j = 0; j < this.points.length; j++) {
        if (i === j) {
          continue;
        }
        if (this.graph[j][i] !== null) {
          this.graph[i][j] = this.graph[j][i];
          this.knnGraph[i][j] = this.graph[i][j];
          continue;
        }
        // weight = 1 / distance
        this.graph[i][j] = 1.0 /
CoordinatesCalculator.getDistance(this.points[i], this.points[j]);
        this.knnGraph[i][j] = this.graph[i][j];
      }
    }
  }

  runKnn() {
    for (let i = 0; i < this.points.length; i++) {
      const weightsSorted = this.knnGraph[i].filter((item) =>
item).reverse();
      const minWeight = weightsSorted[k - 1];

      for (let j = 0; j < this.points.length; j++) {
        if (this.knnGraph[i][j] !== null && this.knnGraph[i][j] <
minWeight) {
          this.knnGraph[i][j] = null;
        }
      }
    }

    for (let i = 0; i < this.points.length; i++) {
      for (let j = 0; j < this.points.length; j++) {
```

```

        if (this.knnGraph[i][j] == null && this.knnGraph[j][i] !=
null) {
            this.knnGraph[i][j] = this.knnGraph[j][i];
        }
    }
}

initClusters() {
    const visitedPoints = new Array(this.points.length).fill(false);

    for (let i = 0; i < visitedPoints.length; i++) {
        if (!visitedPoints[i]) {
            const connectedPoints = runDfs(i, visitedPoints, []);
            const cluster = new Cluster();
            cluster.setPoints(connectedPoints);
            cluster.setGraph(createSubgraph(connectedPoints));
            clusters.add(cluster);
        }
    }
}

runDfs(idx, visitedPoints, connectedPoints) {
    visitedPoints[idx] = true;
    connectedPoints.push(this.points[idx]);

    for (let i = 0; i < visitedPoints.length; i++) {
        if (!visitedPoints[i] && this.knnGraph[idx][i] != null) {
            this.runDfs(i, visitedPoints, connectedPoints);
        }
    }

    return connectedPoints;
}

run() {
    this.initCompleteGraph();
    this.runKnn();
    this.initClusters();
    while (this.clusters.length < this.initNrOfClusters) {
        const clusterToPartition = this.clusters.map()
            .max(Comparator.comparing((c) => c.getPoints().size()));
        const twoClusters = partitionCluster(clusterToPartition);
        this.clusters.push(twoClusters);
        this.clusters.pop();
    }
    while (this.clusters.length > this.resultNrOfClusters) {
        const twoClusters = findTwoClustersToConnect();
        const resultCluster = mergeTwoClusters(twoClusters[0],
twoClusters[1]);
        this.clusters.add(resultCluster);
        this.clusters.removeAll(twoClusters);
    }

    return this.clusters;
}
}

```

ДОДАТОК В

АПРОБАЦІЯ РЕЗУЛЬТАТІВ РОБОТИ

Тези доповіді на міжнародній студентській науковій конференції
«Діджиталізація науки як виклик сьогодення»

Абраменко Роман Олександрович

здобувач вищої освіти факультету комп'ютерних наук

Харківський національний університет радіоелектроніки, Україна

Науковий керівник:

Ревенчук Ілона Анатоліївна, кандидат технічних наук, доцент, доцент кафедри програмної інженерії

Харківський національний університет радіоелектроніки, Україна

ПЕРЕВАГИ ТА НЕДОЛІКИ ОНЛАЙН ОПИТУВАННЯ КОРИСТУВАЧІВ МЕРЕЖ

Дані мають першорядне значення для досліджень. Хоча методологія дослідження може бути різною, кожне дослідження базується на даних, які мають бути якісними і які потім аналізуються та інтерпретуються для отримання інформації. Дослідження переважно ґрунтуються на первинних даних. Найпоширенішим серед кількох методів первинного збору даних є опитування. З використанням ймовірності вибірки в 1930-х роках опитування стали стандартним інструментом для емпіричних досліджень соціальних наук, маркетингу та офіційної статистики. Відповідно до словника Оксфорда, опитування - це дослідження думок чи досвіду групи людей на основі низки запитань. Запитання, що використовуються в опитуванні, як правило, є сукупністю заздалегідь заданих питань, головним чином спрямованих на отримання конкретних даних у певної групи людей щодо їх уподобань, думок, поведінки чи фактичної інформації, залежно від мети опитування. Опитування можуть бути конкретними та обмеженими, або вони можуть мати більш глобальні, широко розповсюджені цілі.

Зокрема, протягом останніх 25-35 років технологія зробила революцію в способі управління опитуваннями - з появою перших опитувань електронної пошти у 1980-х та перших веб-опитуваннях у 1990-х. Так які переваги і недоліки інтернет-опитувань.

Переваги інтернет-опитувань:

- глобальне охоплення. Дає можливість отримання інформації від респондентів, які проживають в різних куточках країни чи по всьому світу, просто та з низькою вартістю;
- гнучкість. Інтернет-опитування є досить гнучкими. Вони можуть бути легко адаптовані до демографічних даних клієнтів, мови, досвіду тощо;
- швидкість та своєчасність. Інтернет-опитування можна проводити з ефективністю часу, мінімізуючи період, необхідний для збору даних;

- зручність. Інтернет-опитування забезпечують зручність кількома способами. Респонденти обирають зручний для себе час і мають стільки часу, скільки потрібно для відповіді на окремі запитання;

- легкість введення та аналізу даних. Для компаній, які проводять онлайн-опитування, значна частина адміністративного тягаря надсилання та отримання анкетування, а також введення даних значно скорочується;

- різноманітність питань. Опитування в Інтернеті можуть включати запитання з множинним вибором, запитання у мультимедійному форматі, питання однозначної і навіть запитання відкритого типу;

- контроль порядку відповідей. На відміну від опитувань електронною поштою, Інтернет-опитування може вимагати від респондента відповідати на запитання у порядку, призначеному дизайнером дослідження, а також заборонити респонденту заздалегідь розглядати подальші запитання. Це зменшує зміщення обстеження;

- обов'язкове заповнення відповідей;

- пристосування опитування до користувача. Опитування в інтернеті можна побудувати так, щоб респонденти відповідали лише на ті питання, які стосуються їх конкретно.

Недоліки інтернет-опитувань:

- до недавнього часу користувачі інтернету не репрезентують загальну кількість населення в усьому світі тому, що доступ до мережі інтернет все ще сильно різниться залежно від країни;

- відсутність досвіду у користувачів. Незважаючи на те, що населення Інтернету стає все більш представницьким, можливо, все ще існують труднощі з опитуваннями через відсутність непоінформованості;

- нечіткі відповіді на запитання. Оскільки огляди в Інтернеті здійснюються самостійно, відповіді на запитання повинні бути гранично чіткими;

- безособовість. В онлайн-опитуваннях зазвичай немає контакту з людиною;

- питання конфіденційності та безпеки. Стандартні опитування електронної пошти не мають високого рівня безпеки. Повідомлення можна перехоплювати. Крім того, багато респондентів задаються питанням, чи відповіді на них будуть розглядатися по-доставірно та чи буде їх контактна інформація продаватися іншим.

Список використаних джерел:

1. Бобровський, М. В. (2019). Абетка для директора. Київ: Державна служба якості освіти України.
2. 10 Key Advantages of Using Online Surveys. *Chanel Sutherland, Explorance. Chanel Sutherland, Explorance*. Вилучено з <https://explorance.com/blog/10-key-advantages-of-using-online-surveys-2-2/>.

Тези доповіді на молодіжному форумі «Радіоелектроніка та молодь у ХХІ ст»

АНАЛІЗ АЛГОРИТМІВ ПОШУКУ В СТРУКТУРІ ДАНИХ ДЛЯ СТВОРЕННЯ ЗРУЧНОГО МЕХАНІЗМУ ОПИТУВАННЯ КОРИСТУВАЧІВ МЕРЕЖ

Абраменко Р.О.

Науковий керівник – доц. Ревенчук І.А.

Харківський національний університет радіоелектроніки
(61166, Харків, просп. Науки, 14, каф. Радіотехніки, тел. (057) 702-00-00)
e-mail: roman.abramenko@nure.ua

Undoubtedly, classification is the basic process in human intellectual activity. Known to us. Considering a group of any objects, we involuntarily divide them into subgroups of closely related elements. Classification is present in the ordering of facts, phenomena and objects known to us. Based on the above, we can conclude that classification is a fundamental concept of science and practice.

Безсумнівно, класифікація – це основний процес в інтелектуальній діяльності людини. Зустрічаючись з новим явищем, ми намагаємося знайти йому аналог у відомій нам області. Розглядаючи групу яких-небудь об'єктів, ми мимоволі розділяємо їх на підгрупи близьких один одному елементів. Класифікація присутня при упорядкуванні відомих нам фактів, явищ, предметів. На підставі сказаного можна зробити висновок, що класифікація – це фундаментальне поняття науки і практики.

Класифікаційний аналіз – це сукупність методів, які дозволяють класифікувати багатомірні спостереження, кожне з яких описується набором вихідних перемінних $X_1, X_2 \dots X_n$. Метою класифікаційного аналізу є утворення груп, схожих між собою об'єктів, що прийнято називати кластерами.

Методи кластерного аналізу

Існує ряд різних методів, які можна використовувати для проведення кластерного аналізу. Ці методи можна класифікувати наступним чином:

1. ієрархічні (агломеративні, дивізійні);
2. неієрархічні (часто відомі як K-методи кластеризації).

Ієрархічні методи

В рамках цього підходу до кластерного аналізу існує ряд різних методів, що використовуються для визначення, які кластери повинні бути об'єднані на кожному етапі. Основні методи узагальнені нижче:

Для моделювання ігрової ситуації були розроблені:
– база даних, которая содержит информацию о пользователях системы, боях между игроками, информацию о всех игровых объектах с их характеристиками (характеристики юнитов и башен) и т.п.;

178

– графова модель для програмного представлення путей розположених на ігровому полі системи, которая позволяет минимизировать движение объектов по карте и поиск путей;
– оптимизационная модель многокритериальной задачи принятия решений для нахождения оптимальных путей для движения юнитов и оптимальных целей для атаки башен.

Для поиска кратчайших путей на карте был спроектирован и реализован алгоритм, основанный на алгоритме Дейкстры нахождения кратчайшего пути между вершинами на графе [1]. Данный алгоритм используется при движении некоторых видов юнитов.

Для поиска оптимального пути нападающего юнита адаптирован алгоритм взвешенной суммы оценок на основании аддитивной свертки с введенными коэффициентами [2], который учитывает следующие критерии:

- общее количество защитных башен на пути;
- безопасность пути: количество единиц урона, которые потенциально могут нанести юниту башни на пути;
- потенциальное количество ресурсов, которые может получить юнит, пройдя по пути или его части;
- отношение защитных башен на пути с особыми эффектами при атаке к общему количеству башен на пути.

В результате была разработана игровая веб-система, которая выполняет следующие основные функции:

- возможность участия пользователя в битве с другим пользователем, установив на карте новых башен, запуска новых монстров;
- автоматический расчет на сервере порядка действий всех игровых объектов на карте для каждого состояния битвы;

– поиск оптимальных путей юнитов и оптимальных целей для атаки башен на основании многокритериальной оптимизационной модели и т.д.

Новизна разработки заключается в наличии математической составляющей, которая позволяет игровым объектам находить оптимальные решения, за счет чего поддерживается более разнообразный геймплей. Кроме того, новизна разработанной системы заключается в возможности сражаться и соревноваться в режиме реального времени с другими игроками, что не распространено среди игр представленного жанра.

Список источников:

1. Волпенская Т.В., Князьков В.С. Компьютерная математика. Часть 2. Теория графов: Учебное пособие. – Пенза: Изд-во Пенз. гос. ун-та, 2002. – 101 с.
2. Лотов А.В., Поспелова И.И. Многокритериальные задачи принятия решений: Учебное пособие. – М.:Издательский отдел факультета ВМиК МГУ им. М.В. Ломоносова; МАКС Пресс, 2008.

179