Юр А.С.
студент,
Наукові керівники: **Сердаковський В.С.**
старший викладач;
Павлов В.А.
старший викладач,
Національний технічний університет України
«Київський політехнічний інститут»

ВЕБ-ДОДАТОК ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАНИХ

Одним із надзвичайно важливих напрямків людської діяльності на сьогодні – аналіз даних, особливо використання найсучасніших знань в області штучного інтелекту. Звідси і почала свій розвиток дисципліна інтелектуального аналізу даних.

Серед ряду задач, які розглядаються дисципліною Data Mining, є кластеризація даних.

Існує багато алгоритмів, які виконують задачу кластеризації вхідних даних, але в такому випадку виникає питання вибору між цими алгоритмами. Проблема складається в тому, що дані можуть володіти різними властивостями, а це в свою чергу призводить диференціації і придатності різних алгоритмів кластеризації.

Метою роботи є створення веб-додатку для впровадження модулів різних алгоритмів аналізу даних (першочергово було обрано щільнісний алгоритм кластеризації даних). Веб-додаток повинен бути у вільному доступі через мережу інтернет, щоб кожен вчений мав змогу дослідити властивості того чи іншого алгиртму та запропонувати варіанти покращення алгоритму або виявити нові особливості його роботи на конкретних прикладах.

Новизна даної роботи полягає у перевазі використання хмарних сервісів для проведення розрахунків та статистичного аналізу даних. Доступність даного сервісу дозволяє швидко, зручно та надійно використовувати різноманітні алгоритми аналізу даних. Кожну нову розробку або істотне покращення існуючого алгоритму можна опублікувати в даному веб-сервісі і тоді кожен вчений зацікавлений в досліджені цього алгоритму зможе без значних затрат перевірити роботу алгоритму, переконатись в нових властивостях.

Також для даної роботи розглядаються сучасні методи та технології розробки веб-додатків та проектування баз даних. Проводиться розгляд існуючих платформ із реалізованими бібліотеками алгоритмамів інтелектуального аналізу даних [4].

Зрозумівши всі переваги та недоліки реляційних баз даних та нереляційних, постає питання вибору однієї із них.

Тому спочатку було запропоновано перелік вимог до бази даних:

– Можливість завантаження будь-якої таблиці із даними в базу даних;
– Швидкість обробки інформації;

– Можливість розширення існуючої таблиці, шляхом створення нових записів;

– Можливість видалення таблиці зі даними;

– Можливість перегляду існуючих таблиць в базі даних;

– Можливість вивантаження таблиці даних з БД як локальний файл на персональний комп'ютер;

Враховуючи, що для поставленої задачі реалізації веб-сервісу немає необхідності в чітко визначеній заздалегіть структурі даних. Тобто, заздалегіть невідомий перелік атрибутів (стовпців, признаків) таблиці з даними, що надає змогу оператору заватажувати та працювати із будь-якою моделю даних. Для такої потреби краще вибрати нереляційну базу даних, яка немає обмежень щодо структури та метаданих таблиці, яка завантажується в сховище [1].

Наступним кроком став вибір бібліотек та фреймворку програмування. Враховуючи специфіку роботи із статистичним аналізом даних та готових бібліотек в кожній із мов програмування, було прийнято рішення зупинитися на R-мові програмування, бо саме *R* почала свій розвиток, як мова статистичних розрахунків. Також в *R* досить розвинені бібліотеки для візуалізації даних. Також *R* представляє із себе повноцінне середовище для інтелектуального аналізу даних, яке має велику спільноту зацікавлених розробників та вчених.

В якості самого фреймворку для розробки веб-додатку вибір зупинився на *Shiny*. Цей вибір аргументований тим, що *RStudio* представила дуже зручну у використані документацію та численні приклади можливостей фреймворку, а також основної функціональності, використовуваної при розробці веб-додатків.

Особливість даного фреймворку полягає також в тому, що з особливостями реалізації, код клієнтської частини фактично генерується без єдиного рядка HTML, CSS коду, а це стає зручним для користувачів, які не мають практичних навичок у розробці графічного інтерфейсу користувача в вебі [5].

Завершальним етапом розробки стало дослідження роботи існуючих реалізацій щільнісного алгоритму кластеризації (Density-based spatial clustering of applications with noise), для того щоб запропонувати покращену версію алгоритму [2].

Найбільшою проблемою існуючої реалізації щільнісного алгоритму кластеризації є те, що алгоритм погано працює з даними де щільність кожного кластеру має різне значення, бо при збільшенні параметру ε росте ймовірність утворення одного великого кластеру замість утворення одного додаткового [3].

Також значних зусиль та часу вимагає вибір значення для параметру ε і не завжди емпірично вдається його коректно підібрати.

Для початку на меті було вирішити завдання автоматичного підбору параметра ε при розробці удосконаленого щільнісного алгоритму кластеризації даних.

Оскільки саме параметр ε в більшій мірі впливає на роботу алгоритму і встановлено, що він може приймати різні значення в залежності від досліджуваної області вхідних даних, було зроблено висновок, що на вхід

алгоритму необхідно ітеративно передавати різні значення цього параметру, а отримані результати аналізувати.

Найбільш доцільним параметром для оцінки результату кластеризації стала кількість самих кластерів, які обов'язково задовільняють первиним вимогам алгоритму, а саме кількість об'єктів в кластері повина бути більшою, аніж параметр MinPts, який за замовчуванням приймає значення на одиницю більше від кількості признаків даних.

В такому випадку після кожної ітерації виконання алгоритму проводилось порівняння із найкращим результатом. Найкращий результат вважався той, який максимізував кількість кластерів при цьому мінімізуючи кількість точок, які важались шумом. Тому даний крок звівся до класичної задачі max-min.

Варто зазначити, що для такої моделі роботи алгритму все-одно необхідно було вирішити питання ініціалізації початковим значенням для ε параметру, а також крок його зміни.

На допомогу прийшов метод розрахунку відстаней між найближчими сусідами kNNdist, окрім вхідних даних ми повині також передати на вхід алгоритму кількість найближчих сусідів, які використовуються (в нашому випадку це мінімальна кількість об'єктів необхідна для утворення кластерів). Таким чином, ми отримуємо матрицю дистанцій, яку згодом перетворюэмо у вектор d.

Емпіричним шляхом було встановлено, що для ініціалізації параметру ε доцільно використовувати:

$$\varepsilon = \mathrm{M}(d) + 2 * \sigma\,(d)$$

Також можна використовувати три стандартних відхилення замість двох, але у всіх досліджуваних випадках двох було цілком достатньо.

Таким чином, було автоматизовано процес підбору параметру ε для алгоритму щільнісної кластеризації даних, а відповідно і вирішено поставлено мету. Дану розробку було додано в веб-додаток та опубліковано в хмарному сервісі.

На зараз веб-додаток доступний по посиланню: https://fbme2015.shinyapps.io/WebApplication/

Список використаних джерел:

1. NoSQL: новая методология разработки нереляционных баз данных: Пер. с англ. – М.: ООО «И.Д. Вильямс», 2013. – 192 с.

2. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise Martin Ester, Hans-Peter Kriegel, Jiirg Sander, Xiaowei Xu. Електроний ресурс: https://www.aaai.org/Papers/KDD/1996/KDD96-037.pdf

3. DBSCAN Revisited: Mis-Claim, Un-Fixability, and Approximation Junhao Gan, Yufei Tao. – Електроний ресурс: http://www.cse.cuhk.edu.hk/~taoyf/paper/sigmod15-dbscan.pdf

4. Guide to Web Application Development Guides, Resources, and Best Practices by Bernard Kohan – Електроний ресурс: http://www.comentum.com/guide-to-web-application-development.html

5. Articles for shiny apps – Електроний ресурс: http://shiny.rstudio.com/articles/