DOI 10.34185/1562-9945-1-132-2021-14
УДК 004.853

К.Ю. Островська, І.В. Стовпченко, В.В. Аніщенко

ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ РОЗПОДІЛЕНИХ АЛГОРИТМІВ МАШИННОГО НАВЧАННЯ

Анотація. Робота присвячена дослідженню ефективності розподілених алгоритмів машинного навчання реалізованих в проекті Apache Mahout.

В результаті роботи був проведений аналіз ефективності алгоритмів машинного навчання за допомогою методу кластеризації к-середніх (k-Means) і методу нечіткої кластеризації к-середніх (fuzzy k-Means / c-Means), реалізованих в проекті Apache Mahout.

Отримано результати тестування обох методів кластеризації на однакових наборах даних.

Розглянуто точність кластеризації кожного методу, а також побудовані порівняльні діаграми результатів досліджуваних методів.

Ключові слова: алгоритм, apache mahout, k-means, fuzzy k-means / c-means, нечітка кластеризація, машинне навчання, hadoop.

У даній роботі розглянуто зберігання, обробка і аналіз великих обсягів даних, а також алгоритми машинного навчання, які реалізують обробку і вилучення необхідної інформації з великих, не завжди структурованих обсягів даних.

Постановка задачі. В роботі потрібно провести аналіз ефективності алгоритмів машинного навчання за допомогою методу кластеризації к-середніх (k-Means) і методу нечіткої кластеризації к-середніх (fuzzy k-Means / c-Means), реалізованих в проекті Apache Mahout.

Провести збірку і настройку кластера, на якому буде виконуватися тестування. На обчислювальному кластері розгорнути розподілену файлову систему HDFS, призначену для обробки великих обсягів даних.

Розгорнути проект Apache Mahout, який надає доступ до реалізації розподілених і масштабованих алгоритмів машинного навчання.

Отримати результати тестування обох методів кластеризації на однакових наборах даних.

Дослідити точність кластеризації кожного методу, а також побудувати порівняльні діаграми результатів досліджуваних методів.

Апаратно-програмне середовище. Hadoop - це вільно поширюваний набір утиліт, бібліотек і фреймворк для розробки і виконання розподілених програм, що працюють на кластерах з сотень і тисяч вузлів. Ця технологія зберігання і обробки Big Data є проектом верхнього рівня фонду Apache Software Foundation.

Проект складається з основних 4-х модулів:

- Hadoop Common - набір програмних бібліотек і утиліт, які використовуються для управління розподіленими файлами і створення необхідної інфраструктури;

- HDFS - розподілена файлова система, Hadoop Distributed File System - технологія зберігання файлів на різних серверах даних;

- MapReduce - набір системних програм, що забезпечують спільне використання, масштабування і надійність роботи розподілених додатків;

- Hadoop MapReduce - платформа програмування і виконання розподілених MapReduce-обчислень, з використанням великої кількості комп'ютерів (вузлів), що утворюють кластер.

Для полегшення машинного навчання на великих даних в Apache Software Foundation працює проект під назвою «Apache Mahout».

Mahout - перша велика бібліотека, що реалізувала багато популярних алгоритми засобами MapReduce.

Мета Apache Mahout - надати масштабовані бібліотеки, які дозволяють розподілено запускати різні алгоритми машинного навчання в Hadoop. На даний момент Mahout підтримує тільки кластеризацию, класифікацію і розробку рекомендацій.

Кластеризація являє з себе процедуру організації елементів в групи, на основі подібності між елементами за певними критеріями.

Для порівняння двох об'єктів, необхідно мати критерій, на підставі якого буде відбуватися порівняння.

Як правило, таким критерієм для алгоритмів кластеризації є відстань між об'єктами.

ISSN 1562-9945 (Print)
ISSN 2707-7977 (Online)

Кластеризація за допомогою Apache Mahout. Бібліотека Mahout підтримує кілька варіантів алгоритмів кластеризації, написаних в парадигмі Map - Reduce, кожен зі своїм власним набором цілей і критеріїв.

k-Means (fuzzy k-Means): групує елементи в k-кластери, ґрунтуючись на відстані від цих елементів до центроїда, або центра ваги попередньої ітерації.

Використовуючи Mahout можна кластеризувати набір даних за допомогою алгоритму k-Means, що складається з наступних етапів:

- Вибір оптимального алгоритму для угруповання елементів кластера, в нашому случає це алгоритм k-Means .

- Визначити метрики для перевірки схожості з знову виявленими елементами.

- Визначити умова зупинки, коли подальша кластеризаці не має сенсу.

- Підготовка вхідних даних.

- Далі необхідно перетворити файли в файли, які розташовані допомогою команди seqdirectory, яка генерує проміжне уявлення документа в форматі SequenceFile з текстових документів в структурі каталогів.

- Перетворення текстових документів в форматі SequenceFile в вектори.

- Запуск обраного алгоритму кластеризації за допомогою одного з безлічі, які підтримують Hadoop програми-драйверів, наявних в Mahout.

- Оцінка одержаних результатів.

- Повторення алгоритму в міру необхідності.

Методи кластеризації, що тестуються. В результаті проведеної роботи були використані два алгоритми кластеризації k-Means і c-Means (Fuzzy k-Means) с метою протестувати дані алгоритми машинного навчання, виявити їх переваги та недоліки, а так само провести порівняльний аналіз результатів.

Метод кластеризації K-Means. Алгоритм роботи k-Means.

1. Здається число кластерів k, яке повинно бути сформовано з об'єктів вихідної вибірки.

2. Випадковим чином вибирається k записів вихідної вибірки, які слугуватимуть початковими центрами кластерів.

3. Для кожного запису вихідної вибірки визначається найближчий до неї центр кластера.

4. Проводиться обчислення - центрів тяжіння кластерів. Це робиться шляхом простого визначення середнього для значень кожного ознаки для всіх записів в кластері.

Кроки 3 і 4 повторюються до тих пір, поки виконання алгоритму лише буде перервано або не буде виконана умова відповідно до деякого критерію збіжності.

Зупинка алгоритму проводиться тоді, коли кордони кластерів і розташування центроїдів не перестануть змінюватися від ітерації до ітерації, тобто на кожній ітерації в кожному кластері буде залишатися один і той же набір записів.

Тестування алгоритму K-Means за допомогою Mahout. Всі об'єкти повинні бути представлені у вигляді набору числових ознак. Користувач повинен вказати кількість k груп, які він бажає ідентифікувати.

Кожен об'єкт може розглядатися як представлений деяким вектором ознак в n -мірному просторі, де n - це число всіх об'єктів, використовуваних для опису об'єктів для кластеризації.

Каталог даних містить кілька вхідних файлів SequenceFile (Key, VectorWritable), каталог кластерів містить один або кілька файлів SequenceFile (Text, Cluster), що містять k вихідних кластерів. Жоден з вхідних пошукових систем не модифікується реалізацією, що дозволяє експериментувати з початковими значеннями кластеризації і збіжності.

Алгоритм кластеризації k-Means може бути запущений з використанням виклику командного рядка для KMeansDriver .main або шляхом виклику Java для KMeansDriver.runJob ().

Діаграма на рисунку 1, показує потік даних прикладу реалізації k-Means, наданого в Mahout.

ISSN 1562-9945 (Print)
ISSN 2707-7977 (Online)

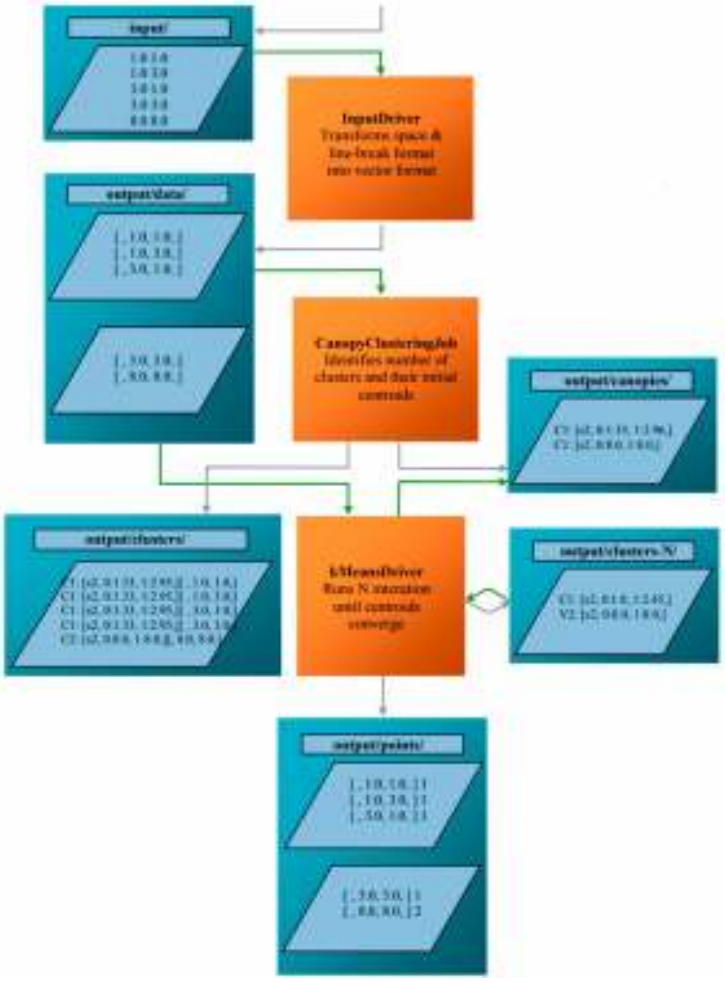

Рисунок1 - Потік даних алгоритму k-Means в Mahout

Результати кластеризації K-Means. Рисунки 2 - 4 ілюструють кластеризацію k-Means, застосовану до набору випадково згенерували двовимірних точок даних.

Точки генеруються з використанням нормального розподілу з центром в середньому місці і з постійним стандартним відхиленням.

Точки генеруються наступним чином:

500 точок m = [1,0, 1,0] sd = 3,0

300 точок m = [1,0, 0,0] sd = 0,5

300 точок m = [0,0, 2,0] sd = 0,1

На рисунку 3 побудовані точки і накладені кордони 3-sigma генератора.

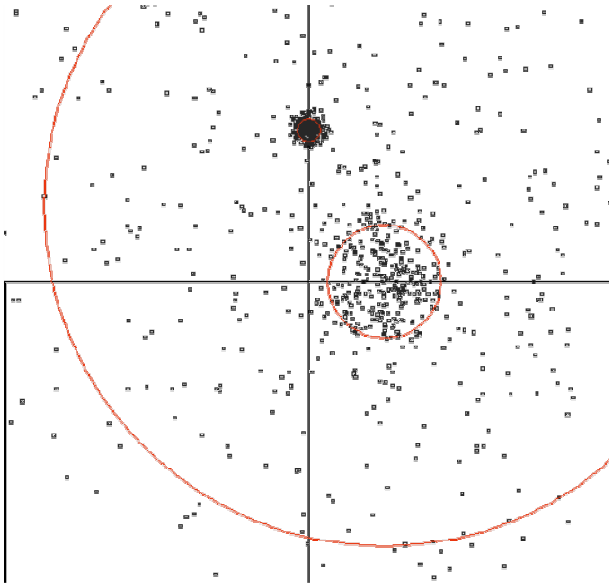

Рисунок 2 - Генеруються точки вибірки

На рисунку 3 результуючі кластери (k=3) показані накладеними на дані вибірки.

Оскільки k-Means є ітеративним алгоритмом, центри кластерів в кожній попередній ітерації показані різними кольорами.

Жирний червоний колір позначає остаточну кластеризацию, а попередні ітерації показані [помаранчевим, жовтим, зеленим, синім, фіолетовим і сірим].

Хоча алгоритм пропускає багато точок і не може захопити вихідні кластерні центри з накладенням, він непогано справляється з кластеризацией цих даних.

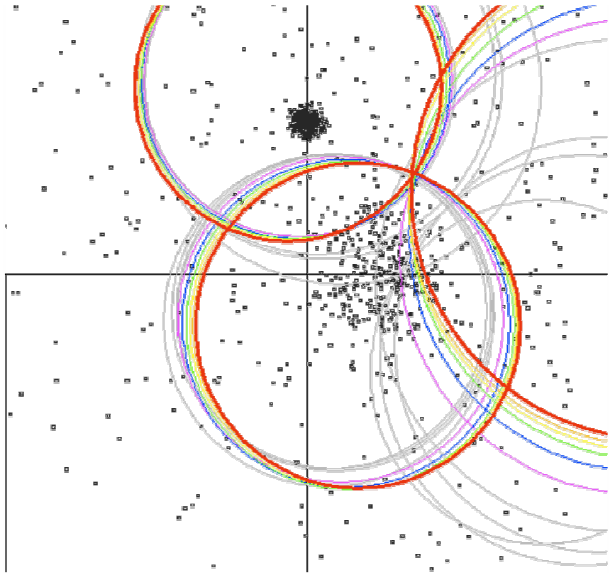

Рисунок 3 - Результуючі кластери k-Means

ISSN 1562-9945 (Print)
ISSN 2707-7977 (Online)

На рисунку 4 показані результати запуску k-Means для іншого набору даних, який генерується з використанням асиметричних стандартних відхилень. k-Means також справляється з цим набором даних.

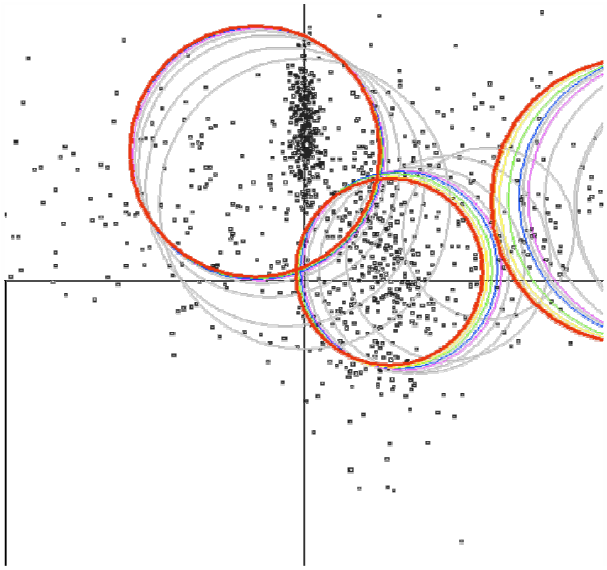

Рисунок 4 - Результати алгоритму k-Means на вибірці,
згенерованої з симетричним стандартним відхиленням

Метод нечіткої кластеризації fuzzy k-Means. Метод нечіткої кластеризації fuzzy k-Means (c-Means) дозволяє розбити наявне безліч елементів потужністю N на задане число нечітких множин k . Метод нечіткої кластеризації c-Means можна розглядати як вдосконалений метод k-Means, при якому для кожного елемента з розглянутої безлічі розраховується ступінь її приналежності кожному з кластерів.

Етапи реалізації алгоритму:

1. Поставити випадковим чином k центрів кластерів c_j, j = 1..k;

2. Розрахувати матрицю приналежності елементів до кластерів r. В разі нормального розподілу:

$$r_{ij} = \frac{\mathcal{N}(d(x_i, c_j)|\mu = 0, \sigma)}{\sum_{j}^{k} \mathcal{N}(d(x_i, c_j)|\mu = 0, \sigma)}, \quad (1)$$

де x_i - i -й елемент безлічі, c_j - центр кластера j, $d(x_i, c_j)$ – відстань між точками x_i і c_j, N - щільність ймовірності нормального розподілу в точці d (x_i, c_j).

3. Перемістити центри кластерів

$$c_j \leftarrow \frac{\sum_i r_{ij} x_i}{\sum_i r_{ij}} \quad (2)$$

4. Розрахувати функцію втрат. У разі нормального розподілу функція втрат буде дорівнює:

$$J = \sum_j^k \sum_i^N d(x_i, c_j)^2 r_{ij} \quad (3)$$

5. Якщо значення функції втрат зменшується, то повторити цикл з пункту 2.

Тестування алгоритма fuzzy k-Means за допомогою Mahout. Fuzzy k-Means (Fuzzy c-Means) є розширенням K- Means, популярного простого методу кластеризації. У той час як k-Means виявляє жорсткі кластери (точки належать тільки одному кластеру), Fuzzy k-Means є більш статистично формалізованим методом і виявляє м'які кластери, в яких конкретна точка може належати більш ніж до одного кластеру з певною ймовірністю.

Як і k-Means, Fuzzy k-Means працює з тими об'єктами, які можуть бути представлені в n - мірному векторному просторі, і визначається міра відстані.

Алгоритм:

1. Ініціалізувати k кластерів

2. Поки не зійшлися:

- Обчислити ймовірність того, що точка належить кластеру для кожної пари <point, cluster>.

- Перерахувати центри кластерів, використовуючи наведені вище значення ймовірності членства точок в кластерах.

Реалізація алгоритму fuzzy K-Means. Він приймає вхідний файл, який містить векторні точки. Користувач може надати центри кластерів в якості вхідних даних або дозволити запуск алгоритму навісу і створення початкових кластерів.

Подібно k-Means, програма не змінює вхідні каталоги. І для кожної ітерації вихідні дані кластера зберігаються в каталозі cluster-N.

Алгоритм Fuzzy k-Means може бути запущений з використанням виклику командного рядка для FuzzyKMeansDriver.main або шляхом виклику Java для FuzzyKMeansDriver.run().

ISSN 1562-9945 (Print)
ISSN 2707-7977 (Online)

Результати кластеризації методу fuzzy K-Means. На рисунках 5 – 7 представлені зображення, що ілюструють кластеризацию Fuzzy k-Means, застосовану до набору випадково згенерували двовимірних точок даних.

Точки генеруються з використанням нормального розподілу з центром в середньому місці і з постійним стандартним відхиленням.

Точки генеруються наступним чином:

1. 500 точок m = [1.0, 1.0] sd = 3.0
2. 300 точок m = [1.0, 0.0] sd = 0.5
3. 300 точок m = [0,0, 2.0] sd = 0.1

На рисунку 5 побудовані точки і накладені кордони 3 sigma генератора.

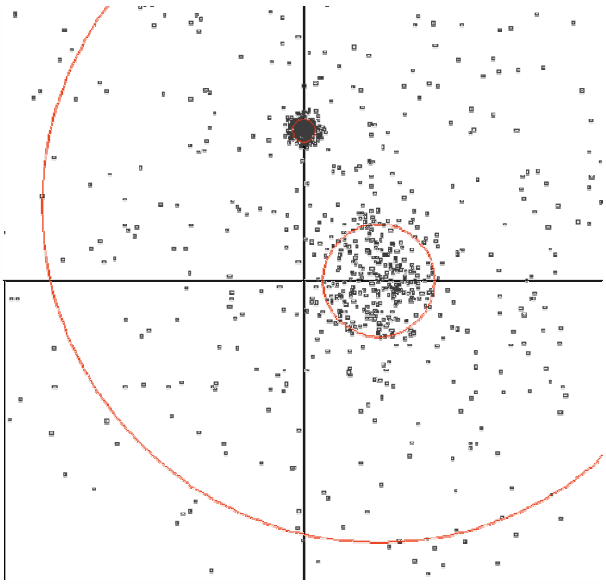

Рисунок 5 - Генеруються точки вибірки

На рисунку 6 результуючі кластери (k = 3) показані накладеними на дані вибірки.

Оскільки Fuzzy k-Means є ітеративним алгоритмом, центри кластерів в кожній недавньої ітерації показані різними кольорами.

Жирний червоний колір позначає остаточну кластеризацию, а попередні ітерації показані [помаранчевим, жовтим, зеленим, синім, фіолетовим і сірим].

Хоча він пропускає багато точок і не може захопити вихідні кластерні центри з накладенням, він непогано справляється з кластеризацией цих даних.

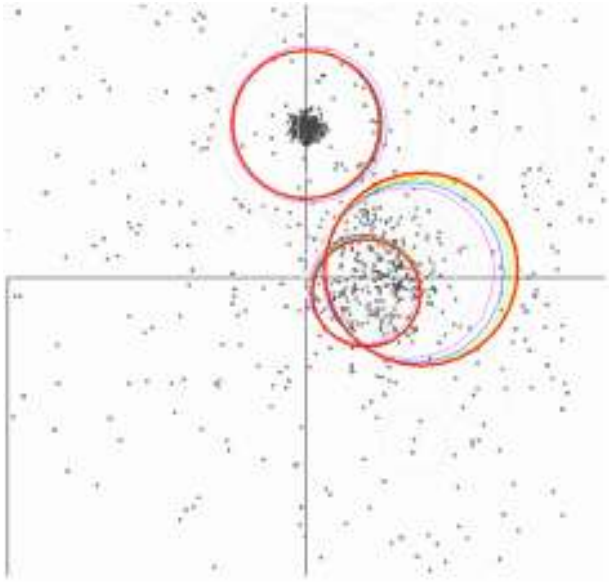

Рисунок 6 - Результуючі кластери fuzzy k-Means

На рисунку 7 показані результати запуску fuzzy k-Means для іншого набору даних, який генерується з використанням асиметричних стандартних відхилень.

Fuzzy k-Means також справляється з цим набором даних.

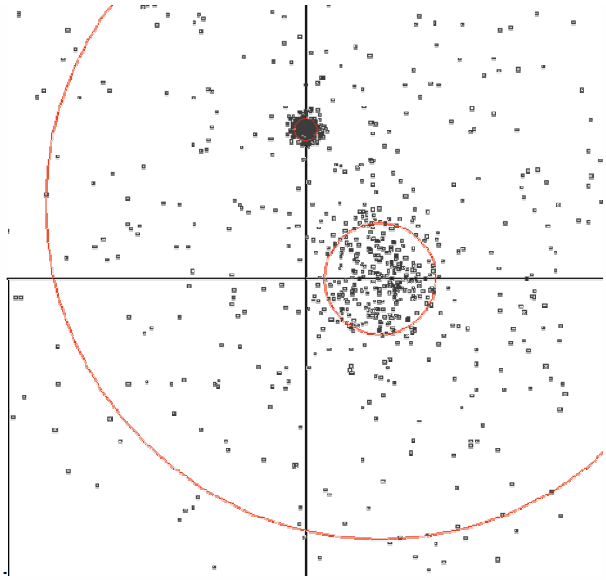

Рисунок 7 - Результати алгоритму fuzzy k-Means на вибірці, згенерованої з симетричним стандартним відхиленням

Конфігурація тестових обчислювальних систем. При тестуванні методів використовувався кластер з двома обчислювальними вузлами на базі платформи Intel Sandy Bridge (таблиця 1).

ISSN 1562-9945 (Print)
ISSN 2707-7977 (Online)

Конфігурація керуючого вузла (таблиця 2).

Для первинного тестування використовувалася одна машина з налаштованим фреймворком Hadoop і Apache Mahout (таблиця) 3.

Таблиця 1

Конфігурація обчислювального вузла

Системная плата	Gigabyte GA-Z77-DS3H
Процессор	QuadCore Intel Core i5-2500K, 4200 MHz (42 x 100)
Оперативная память	16 GB (2 x 8GB DDR3 1600 MHz SDRAM)
Жесткий диск	SATAIII 2000GB (Seagate Barracuda ST2000318AS)
Сетевая карта	Qualcomm Atheros AR8151 PCI-E Gigabit Ethernet Controller (NDIS 6.20)

Таблиця 2

Конфігурація керуючого вузла

Системная плата	DAZAAMB16E0
Процессор	QuadCore Intel Core i5 7200U 3100 МГц
Оперативная память	10 GB (8GB + 2GB DDR4 2133 MHz SODIMM)
SSD диск	Kingston 120Gb VU400
Жесткий диск	SATAIII 1000GB (Western Digital Blue)
Сетевая карта	Qualcomm Atheros AR8151 PCI-E Gigabit Ethernet Controller (NDIS 6.20)

Таблиця 3

Конфігурація локальної ВС

Системная плата	Asus P8Z77-V LK
Процессор	QuadCore Intel Core i7-3770K, 4700 MHz (47 x 100)
Оперативная память	16Gb (2 x 8Gb DDR3 1866 MHz KHX2133C11D3/8GX)
SSD диск	Диск #1 - Samsung SSD 850 EVO 250GB (232 ГБ)
Жесткий диск	Диск #2 - ST2000DM001-1CH164 (1863 ГБ)
Сетевая карта	Realtek RTL8168/8111 Gigabit Ethernet

Висновки. В результаті роботи був проведений аналіз ефективності алгоритмів машинного навчання за допомогою методу кластеризації к-середніх (k-Means) і методу нечіткої кластеризації к-середніх (fuzzy k-Means / c-Means), реалізованих в проекті Apache Mahout.

Отримано результати тестування обох методів кластеризації на однакових наборах даних.

Розглянуто точність кластеризації кожного методу, а також побудовані порівняльні діаграми результатів досліджуваних методів.

ЛІТЕРАТУРА / ЛИТЕРАТУРА

1. Ralf Lammel. Google's MapReduce Programming Model - Revisited. 2017p., 42c. URL: https://userpages.uni-koblenz.de/~laemmel/MapReduce/paper.pdf

2. Tom White. Hadoop: The Definitive Guide. THIRD EDITION. O'RELLY. - 2013p., 647c.

3. Чак Лем. Hadoop в дії. Москва. - 2012р., 448с.
4. MicheleNemschoff. Maximize Performance and Scalability Within Your Hadoop Architecture. 2014p.
URL: https://www.smartdatacollective.com/how-maximize-performance-and-scalability-within-your-hadoop-architecture /
5. Sea Owen, Robin Anil, Ted Dunning, Ellen Friedman. Mahout in action. MANNING. - 2012p., 341c.
6. Adam Coates, Andrew Y. Ng. Learning Feature Representations with K-means, Stanford University, 2012p., 20 стор.
URL: https://cs.stanford.edu/~acoates/papers/coatesng_nntot2012.pdf
7. Єршов К.С., Романова Т.Н. Аналіз і класифікація алгоритмів кластеризації. МГТУ ім. Н.е. Баумана. 2016р. 6с.
8. Tutorial spoint. Mahout - Clustering.
URL: https://www.tutorialspoint.com/mahout/mahout_clustering.htm
9. Alexander N. Gorban, Andrei Y. Zinovyev. Principal Graphs and Manifolds. University of Leicester. 36c.
URL: https://arxiv.org/ftp/arxiv/papers/0809/0809.0490.pdf
10. Kwok, T., Smith, K., Lozano, S., Taniar. Parallel Fuzzy c-Means Clustering for Large Data Sets, 2012p.
URL: http://num-meth.srcc.msu.ru/zhurnal/tom_2012/pdf/v13r207.pdf
11. Михалев А.И., Винокурова Е.А., Сотник С.Л. Компьютерные методы интеллектуальной обработки данных: учебное пособие. – Днепропетровск: НМетАУ, ИК "Системные технологии", 2014. – 209 стр.

REFERENCES

1. Ralf Lammel. Google's MapReduce Programming Model - Revisited. 2017p., 42c.
URL: https://userpages.uni-koblenz.de/~laemmel/MapReduce/paper.pdf
2. Tom White. Hadoop: The Definitive Guide. THIRD EDITION. O'RELLY. - 2013p., 647c.
3. Чак Лем. Hadoop в дії. Москва. - 2012р., 448с.
4. MicheleNemschoff. Maximize Performance and Scalability Within Your Hadoop Architecture. 2014p.
URL: https://www.smartdatacollective.com/how-maximize-performance-and-scalability-within-your-hadoop-architecture /
5. Sea Owen, Robin Anil, Ted Dunning, Ellen Friedman. Mahout in action. MANNING. - 2012p., 341c.

ISSN 1562-9945 (Print)
ISSN 2707-7977 (Online)

6. Adam Coates, Andrew Y. Ng. Learning Feature Representations with K-means, Stanford University, 2012p., 20 стор.
URL: https://cs.stanford.edu/~acoates/papers/coatesng_nntot2012.pdf
7. Ershov K.S., Romanova T.N. Analysis and classification of clustering algorithms. MSTU. Not. Bauman. 2016 6s.
8. Tutorial spoint. Mahout - Clustering.
URL: https://www.tutorialspoint.com/mahout/mahout_clustering.htm
9. Alexander N. Gorban, Andrei Y. Zinovyev. Principal Graphs and Manifolds. University of Leicester. 36c.
URL: https://arxiv.org/ftp/arxiv/papers/0809/0809.0490.pdf
10. Kwok, T., Smith, K., Lozano, S., Taniar. Parallel Fuzzy c-Means Clustering for Large Data Sets, 2012p.
URL: http://num-meth.srcc.msu.ru/zhurnal/tom_2012/pdf/v13r207.pdf
11. Mikhalev A.I., Vinokurova E.A., Sotnik S.L. Computer methods of intelligent data processing: a textbook. - Dnepropetrovsk: NMetAU, IC "System Technologies", 2014. - 209 pages.

Received 01.02.2021.
Accepted 03.02.2021.

Исследование эффективности распределенных алгоритмов машинного обучения

Работа посвящена исследованию эффективности распределенных алгоритмов машинного обучения реализованных в проекте Apache Mahout.

В результате работы был проведен анализ эффективности алгоритмов машинного нав-жение с помощью метода кластеризации к-средних (k-Means) и метода нечеткой кластеризации к-средних (fuzzy k-Means / c-Means), реализованных в проекте Apache Mahout.

Получены результаты тестирования обоих методов кластеризации на одинаковых наборах данных.

Рассмотрены точность кластеризации каждого метода, а также построены сравнительные диаграммы результатов исследуемых методов.

Research of the efficiency of distributed algorithms for machine learning

This paper discusses the storage, processing and analysis of large amounts of data, as well as machine learning algorithms that implement the processing and extraction of the necessary information from large, not always structured amounts of data.

The work is devoted to the study of the effectiveness of distributed machine learning algorithms implemented in the Apache Mahout project.

As a result of the work, an analysis of the effectiveness of machine-guided algorithms was carried out using the k-Means clustering method and the fuzzy k-Means / c-Means method, implemented in the Apache Mahout project.

The results of testing both clustering methods on the same data sets are obtained.

The accuracy of clustering of each method is considered, and comparative diagrams of the results of the investigated methods are constructed.

Островська Катерина Юріївна – к.т.н., доцент, доцент кафедри інформаційних технологій та систем, Національна металургійна академія України.

Стовпченко Іван Володимирович – старший викладач кафедри інформаційних технологій та систем, Національна металургійна академія України.

Аніщенко Владислав Володимирович - магістр кафедри інформаційних технологій та систем, Національна металургійна академія України.

Островская Екатерина Юрьевна - к.т.н., доцент, доцент кафедры информационных технологий и систем, Национальная металлургическая академия Украины.

Стовпченко Иван Владимирович - старший преподаватель кафедры информационных технологий и систем, Национальная металлургическая академия Украины.

Анищенко Владислав Владимирович - магистр кафедры информационных технологий и систем, Национальная металлургическая академия Украины.

Ostrovskaya Ekaterina Yurievna - Candidate of Technical Sciences, Associate Professor, Associate Professor of the Department of Information Technologies and Systems, National Metallurgical Academy of Ukraine.

Stovpchenko Ivan Vladimirovich - Senior Lecturer, Department of Information Technologies and Systems, National Metallurgical Academy of Ukraine.

Anischenko Vladislav Vladimirovich - Master of Information Technologies and Systems Department, National Metallurgical Academy of Ukraine.

ISSN 1562-9945 (Print)
ISSN 2707-7977 (Online)