

Міністерство освіти і науки України
Національний технічний університет України „КПІ”
Факультет інформатики та обчислювальної техніки

Кафедра автоматизованих систем обробки
інформації та управління

ЗВІТ

з лабораторної роботи № 4
дисципліни
“ТЕХНОЛОГІЇ ПАРАЛЕЛЬНОГО ПРОГРАМУВАННЯ В УМОВАХ
ВЕЛИКИХ ДАНИХ”
на тему:

„Реалізація на основі Apache Spark”

Виконали:
студенти групи ІТ-01мн
Панасюк Станіслав
Лесогорський Кирило

Перевірив:
доц. Жереб К. А.

Київ – 2021

Зміст

1. Постановка задачі	3
2. Обрані інструменти.....	3
3. Опис роботи програмного забезпечення.....	3
4. Отримані результати	4
5. Висновки	5

1. Постановка задачі

Обрану задачу необхідно реалізувати, використовуючи Apache Spark. У якості задачі було обрано підготовку зображень до поглинення через Apache Hadoop задля вирішення тої ж задачі, що і у перших лабораторних роботах. У ядрі системи лежатиме використання D-hash для знаходження хешу зображення. D-hash дозволяє точно та швидко шукати схожі зображення. Він стійкий до скейлінгу зображення, але погано справляється з обрізаними та повернутими під кутом зображеннями. Тому цю техніку аугментовано за допомогою наступного прийому: при завантаженні зображення воно буде аугментовано за допомогою декількох фільтрів, при цьому для кожного фільтру буде згенеровано хеш і збережено у базу даних. При пошуку зображення буде використовуватись оператор XOR для знаходження зображень зі схожими хешами.

2. Обрані інструменти

Для виконання четвертої лабораторної роботи буде використано стандартні інструменти Java із доданою до них бібліотекою Apache Spark. У нашому випадку третя та четверта лабораторні роботи взаємопов'язані, адже четверта робота підготовує зображення, використовуючи Apache Spark, а вже третя робота працює із ними за рахунок написаної MapReduce програми.

3. Опис роботи програмного забезпечення

У ході виконання даної лабораторної роботи було створено додаткову логіку у додатку із першою та другою лабораторними, що допомагає третій лабораторній ефективно опрацьовувати зображення.

Сама ж логіка працює у декілька кроків:

1. Налаштування `JavaSparkContext`. Бібліотека, яку ми використовуємо, дозволяє запустити Apache Spark без додаткового встановлення його на комп'ютері, чим ми і скористались. У реальному проєкті Spark був би запуснений на віддаленому сервері та допомагав ефективно паралелити задачі між багатьма серверами;
2. Зчитування зображень у зручний для Apache Spark формат. Ми зчитуємо зображення, як бінарні файли та складаємо їх до `JavaPairRDD`, де ключом є ім'я зображення, а значенням – потік бінарних даних, що є тілом зображення;
3. Записуємо дані у зручний для Apache Hadoop формат. На цьому кроці ми, використовуючи `SequenceFile.Writer` записуємо зображення один за одним

у єдиний SequenceFile, що складається з ключів та значень (у нашому випадку, з імен та бінарних тіл зображень);

4. Отримані результати


В результаті виконання роботи було отримано наступні результати з процесінгу:

```

21/12/21 22:49:08 INFO SparkContext: Created broadcast 2 from DAGScheduler.scala:1427
21/12/21 22:49:08 INFO DAGScheduler: Submitting 1 missing tasks from ResultStage 1 (File:///Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images BinaryFileRDD[0] at binaryFiles at SparkDataTransformer.java:39) (first 15 tasks are for partitions Vector(0))
21/12/21 22:49:08 INFO TaskSchedulerImpl: Adding task set 1.0 with 1 tasks resource profile 0
21/12/21 22:49:08 INFO TaskSetManager: Starting task 0.0 in stage 1.0 (TID 1) (192.168.31.2, executor driver, partition 0, PROCESS_LOCAL, 11938 bytes)
taskResourceAssignments Map()
21/12/21 22:49:08 INFO Executor: Running task 0.0 in stage 1.0 (TID 1)
21/12/21 22:49:08 INFO BinaryFileRDD: Input split: Path=Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/47-test.jpg;0+130110,
/Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/46-test.jpg;0+52017,Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/38-test.jpg;0+47607,
/Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/31-test.jpg;0+92495,Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/50-test.jpg;0+49863,
/Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/1-test.jpg;0+51260,Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/19-test.jpg;0+46689,
/Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/18-test.jpg;0+107802,Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/27-test.jpg;0+54613,
/Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/26-test.jpg;0+39732,Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/13-test.jpg;0+90504,
/Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/12-test.jpg;0+51435,Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/6-test.jpg;0+94400,
/Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/17-test.jpg;0+93493,Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/20-test.jpg;0+70813,
/Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/21-test.jpg;0+82157,Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/14-test.jpg;0+12246,
/Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/15-test.jpg;0+87963,Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/40-test.jpg;0+51164,
/Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/41-test.jpg;0+65367,Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/37-test.jpg;0+122749,
/Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/36-test.jpg;0+51435,Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/30-test.jpg;0+119087,
/Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/11-test.jpg;0+36414,Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/8-test.jpg;0+73141,
/Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/9-test.jpg;0+84097,Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/2-test.jpg;0+88422,
/Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/3-test.jpg;0+80918,Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/24-test.jpg;0+126300,
/Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/25-test.jpg;0+110092,Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/39-test.jpg;0+42034,
/Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/38-test.jpg;0+49765,Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/44-test.jpg;0+107856,
/Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/45-test.jpg;0+45794,Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/33-test.jpg;0+87779,
/Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/32-test.jpg;0+496123,Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/49-test.jpg;0+168214,
/Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/48-test.jpg;0+72411,Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/43-test.jpg;0+160347,
/Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/42-test.jpg;0+59906,Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/34-test.jpg;0+130110,
/Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/35-test.jpg;0+72889,Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/37-test.jpg;0+173396,
/Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/46-test.jpg;0+82653,Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/29-test.jpg;0+134682,
/Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/28-test.jpg;0+26298,Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/5-test.jpg;0+43127,
/Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/4-test.jpg;0+115941,Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/23-test.jpg;0+128158,
/Users/spasniuk/IdeaProjects/KPI/Imrec/TestData/Images/22-test.jpg;0+108155
21/12/21 22:49:08 INFO CodeCommitPool: Got brand-new compressor [b.22]
21/12/21 22:49:09 INFO BlockManagerInfo: Removed broadcast_1_piece0 on 192.168.31.2:58707 in memory (size: 2.2 KiB, free: 127.2 MiB)
21/12/21 22:49:13 INFO Executor: Finished task 0.0 in stage 1.0 (TID 1). 1055 bytes result sent to driver
21/12/21 22:49:13 INFO TaskSetManager: Finished task 0.0 in stage 1.0 (TID 1) in 4785 ms on 192.168.31.2 (executor driver) (1/1)
21/12/21 22:49:13 INFO TaskSchedulerImpl: Removed TaskSet 1.0, whose tasks have all completed, from pool
21/12/21 22:49:13 INFO DAGScheduler: ResultStage 1 (foreachPartition at SparkDataTransformer.java:41) finished in 4.821 s
21/12/21 22:49:13 INFO DAGScheduler: Job 1 is finished. Cancelling potential speculative or zombie tasks for this job
21/12/21 22:49:13 INFO TaskSchedulerImpl: Killing all running tasks in stage 1: Stage finished
21/12/21 22:49:13 INFO DAGScheduler: Job 1 finished: foreachPartition at SparkDataTransformer.java:41, took 4.831145 s

```

Рисунок 4.1 – Частина логів виконання задачі


3.2.0

[Jobs](#) |
 [Stages](#) |
 [Storage](#) |
 [Environment](#) |
 [Executors](#)

Imrec Parser application UI

Spark Jobs (?)

User: spanasiuk
 Total Uptime: 8 s
 Scheduling Mode: FIFO
 Active Jobs: 1
 Completed Jobs: 1

[▶ Event Timeline](#)

[~ Active Jobs \(1\)](#)

Page: 1

1 Pages. Jump to 1. Show 100 items in a page. [Go](#)

Job Id ▾	Description	Submitted	Duration	Stages: Succeeded/Total	Tasks (for all stages): Succeeded/Total
1	forechPartition at SparkDataTransformer.java:41 forechPartition at SparkDataTransformer.java:41	2021/12/21 22:49:08 (kill)	4 s	0/1	0/1

Page: 1

1 Pages. Jump to 1. Show 100 items in a page. [Go](#)

[~ Completed Jobs \(1\)](#)

Page: 1

1 Pages. Jump to 1. Show 100 items in a page. [Go](#)

Job Id ▾	Description	Submitted	Duration	Stages: Succeeded/Total	Tasks (for all stages): Succeeded/Total
0	isEmpty at SparkDataTransformer.java:40 isEmpty at SparkDataTransformer.java:40	2021/12/21 22:49:08	0.4 s	1/1	1/1

Page: 1

1 Pages. Jump to 1. Show 100 items in a page. [Go](#)

Рисунок 4.2 - UI Apache Spark, що показує стан поточних задач

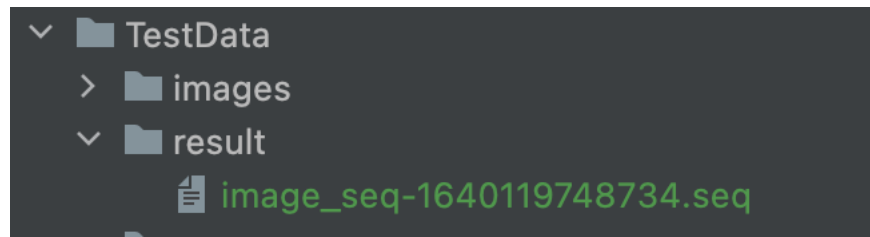


Рисунок 4.3 - Результат опрацювання зображень

5. Висновки

У ході лабораторної роботи було створено додаток із використанням Apache Spark для підготовки даних з ціллю подальшого їх поглинення через Apache Hadoop