

Звіт до лабораторної роботи №3

Розмірковуючи над алгоритмом для реалізації множення матриць з використанням Hadoop MapReduce найважче було зрозуміти як правильно сформувати ключ-значення для reducer-а, щоб потім мати усі потрібні дані для виконання множення та сумування.

Але після того як намалював собі для візуального прикладу множення матриць, із зазначенням індексів (рядів-колонок). Знайшов доволі просту й помітну закономірність між значенням ключів та значень.

The image shows handwritten notes on grid paper illustrating matrix multiplication and the corresponding key-value pairs for a MapReduce implementation.

Matrix Multiplication:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}_{\substack{0,0 & 0,1 \\ 1,0 & 1,1}} \times \begin{bmatrix} e & f \\ g & h \\ i & j \end{bmatrix}_{\substack{0,0 & 0,1 & 0,2 \\ 1,0 & 1,1 & 1,2 \\ 2,0 & 2,1 & 2,2}} = \begin{bmatrix} a \times e + b \times g & a \times f + b \times h \\ c \times e + d \times g & c \times f + d \times h \end{bmatrix}_{\substack{0,0 & 0,1 \\ 1,0 & 1,1}}$$

Index Logic:

index: for A: [row, 0...n]
else B: [column, 0...n]

Value Logic:

value: "A - column - value"
"B - column row - value"

Key-Value Mapping Examples:

$\begin{pmatrix} a & b \\ e & g \end{pmatrix}_{0,0}$	$\begin{pmatrix} a & b \\ + \end{pmatrix}_{0,1}$	$\begin{pmatrix} a & b \\ m \end{pmatrix}_{0,2}$	A-0-e
$\begin{pmatrix} c & d \\ e & g \end{pmatrix}_{1,0}$	$\begin{pmatrix} c & d \\ + \end{pmatrix}_{1,1}$	$\begin{pmatrix} c & d \\ m \end{pmatrix}_{1,2}$	B-0-e

Далі потрібно було тільки імплементувати алгоритм, основні кроки:

1. Створити конфігурацію з вхідних аргументів:

```
// Set dimensions from args
String[] infoTupleA = args[1].split(argsDimensionSeparator);
rowA = Integer.parseInt(infoTupleA[1]);
columnA = Integer.parseInt(infoTupleA[2]);
String[] infoTupleB = args[3].split(argsDimensionSeparator);
columnB = Integer.parseInt(infoTupleB[2]);

conf.setInt(rowAConf, rowA);
conf.setInt(columnAConf, columnA);
conf.setInt(columnBConf, columnB);
```

2. Створення ключа та значення в залежності яка матриця прийшла на вхід:

```
String[] positionAndValue = value.toString().split(positionValueSeparator);
int val = Integer.parseInt(positionAndValue[1]);
String[] rowAndColumn = positionAndValue[0].split(rowColumnSeparator);
int row = Integer.parseInt(rowAndColumn[0]);
int column = Integer.parseInt(rowAndColumn[1]);

if (matrix.equals(firstMatrix)) {
    for (int k = 0; k < columnB; k++) {
        mapKey.set(row + rowColumnSeparator + k);
        mapValue.set(firstMatrix + valueSeparator + column + valueSeparator + val);
        context.write(mapKey, mapValue);
    }
} else {
    for (int i = 0; i < rowA; i++) {
        mapKey.set(i + rowColumnSeparator + column);
        mapValue.set(secondMatrix + valueSeparator + row + valueSeparator + val);
        context.write(mapKey, mapValue);
    }
}
```

3. Розбір значення в редюсері, множення та сумування для знаходження Rik.

```
int sum = 0;
int[] A = new int[columnA];
int[] B = new int[columnA];

for (Text val : values) {
    String[] MatrixIndexValue = val.toString().split(valueSeparator);
    if (MatrixIndexValue[0].equals(firstMatrix)) {
        A[Integer.parseInt(MatrixIndexValue[1])] = Integer.parseInt(MatrixIndexValue[2]);
    } else
        B[Integer.parseInt(MatrixIndexValue[1])] = Integer.parseInt(MatrixIndexValue[2]);
}

for (int j = 0; j < columnA; j++) {
    sum += A[j] * B[j];
}
context.write(key, new Text(Integer.toString(sum)));
sum = 0;
```

Приклад виконання програми з вхідними параметрами:

./input/A.txt A-5-5 ./input/B.txt B-5-5 ./output

Приклад файлу з матрицею A:

A.txt	B.txt	part-r-00000	MatrixMultiply.java
1	0,0 1		
2	0,1 6		
3	0,2 1		
4	0,3 2		
5	0,4 7		
6	1,0 3		
7	1,1 2		
8	1,2 1		
9	1,3 12		
10	1,4 6		
11	2,0 32		
12	2,1 1		
13	2,2 2		
14	2,3 5		
15	2,4 8		
16	3,0 9		
17	3,1 2		
18	3,2 1		
19	3,3 1		
20	3,4 3		
21	4,0 4		
22	4,1 5		
23	4,2 7		
24	4,3 7		
25	4,4 8		

Приклад файлу з матрицею B:

A.txt	B.txt	part-r-00000	MatrixMultiply.java
1	0,0 1		
2	0,1 5		
3	0,2 7		
4	0,3 1		
5	0,4 2		
6	1,0 3		
7	1,1 5		
8	1,2 7		
9	1,3 9		
10	1,4 2		
11	2,0 8		
12	2,1 7		
13	2,2 1		
14	2,3 3		
15	2,4 5		
16	3,0 8		
17	3,1 5		
18	3,2 3		
19	3,3 2		
20	3,4 1		
21	4,0 8		
22	4,1 9		
23	4,2 2		
24	4,3 1		
25	4,4 2		

Очікуваний результат після множення матриць:

$$\begin{pmatrix} 1 & 6 & 1 & 2 & 7 \\ 3 & 2 & 1 & 12 & 6 \\ 32 & 1 & 2 & 5 & 8 \\ 9 & 2 & 1 & 1 & 3 \\ 4 & 5 & 7 & 7 & 8 \end{pmatrix} \cdot \begin{pmatrix} 1 & 5 & 7 & 1 & 2 \\ 3 & 5 & 7 & 9 & 2 \\ 8 & 7 & 1 & 3 & 5 \\ 8 & 5 & 3 & 2 & 1 \\ 8 & 9 & 2 & 1 & 2 \end{pmatrix} = \begin{pmatrix} 99 & 115 & 70 & 69 & 35 \\ 161 & 146 & 84 & 54 & 39 \\ 155 & 276 & 264 & 65 & 97 \\ 55 & 94 & 87 & 35 & 34 \\ 195 & 201 & 107 & 92 & 76 \end{pmatrix}$$

Результуючий файл:

```
A.txt x B.txt x part-r-00000 x MatrixMultiply.java x
1 0,0 99
2 0,1 115
3 0,2 70
4 0,3 69
5 0,4 35
6 1,0 161
7 1,1 146
8 1,2 84
9 1,3 54
10 1,4 39
11 2,0 155
12 2,1 276
13 2,2 264
14 2,3 65
15 2,4 97
16 3,0 55
17 3,1 94
18 3,2 87
19 3,3 35
20 3,4 34
21 4,0 195
22 4,1 201
23 4,2 107
24 4,3 92
25 4,4 76
```

Приклад логів:

```
22/11/29 01:24:00 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
22/11/29 01:24:00 WARN mapred.JobClient: No job jar file set. User classes may not be found. See JobConf(Class) or JobConf#setJar(String).
22/11/29 01:24:00 INFO input.FileInputFormat: Total input paths to process : 2
22/11/29 01:24:00 WARN snappy.LoadSnappy: Snappy native library not loaded
22/11/29 01:24:00 INFO mapred.JobClient: Running job: job_local255255534_0001
22/11/29 01:24:00 INFO mapred.LocalJobRunner: Waiting for map tasks
22/11/29 01:24:00 INFO mapred.LocalJobRunner: Starting task: attempt_local255255534_0001_m_000000_0
22/11/29 01:24:00 INFO mapred.Task: Using ResourceCalculatorPlugin : null
22/11/29 01:24:00 INFO mapred.MapTask: Processing split: file:/Users/mixei1/Learning/BigData/HadoopLabs/input/A.txt:0+151
22/11/29 01:24:00 INFO mapred.MapTask: io.sort.mb = 100
22/11/29 01:24:00 INFO mapred.MapTask: data buffer = 79691776/99614720
22/11/29 01:24:00 INFO mapred.MapTask: record buffer = 262144/327680
22/11/29 01:24:00 INFO mapred.MapTask: Starting flush of map output
22/11/29 01:24:00 INFO mapred.MapTask: Finished spill 0
22/11/29 01:24:00 INFO mapred.Task: Task:attempt_local255255534_0001_m_000000_0 is done. And is in the process of committing
22/11/29 01:24:00 INFO mapred.LocalJobRunner:
22/11/29 01:24:00 INFO mapred.Task: Task 'attempt_local255255534_0001_m_000000_0' done.
22/11/29 01:24:00 INFO mapred.LocalJobRunner: Finishing task: attempt_local255255534_0001_m_000000_0
22/11/29 01:24:00 INFO mapred.LocalJobRunner: Starting task: attempt_local255255534_0001_m_000001_0
22/11/29 01:24:00 INFO mapred.Task: Using ResourceCalculatorPlugin : null
22/11/29 01:24:00 INFO mapred.MapTask: Processing split: file:/Users/mixei1/Learning/BigData/HadoopLabs/input/B.txt:0+149
22/11/29 01:24:00 INFO mapred.MapTask: io.sort.mb = 100
22/11/29 01:24:00 INFO mapred.MapTask: data buffer = 79691776/99614720
22/11/29 01:24:00 INFO mapred.MapTask: record buffer = 262144/327680
22/11/29 01:24:00 INFO mapred.MapTask: Starting flush of map output
22/11/29 01:24:00 INFO mapred.MapTask: Finished spill 0
22/11/29 01:24:00 INFO mapred.Task: Task:attempt_local255255534_0001_m_000001_0 is done. And is in the process of committing
22/11/29 01:24:00 INFO mapred.LocalJobRunner:
22/11/29 01:24:00 INFO mapred.Task: Task 'attempt_local255255534_0001_m_000001_0' done.
22/11/29 01:24:00 INFO mapred.LocalJobRunner: Finishing task: attempt_local255255534_0001_m_000001_0
22/11/29 01:24:00 INFO mapred.LocalJobRunner: Map task executor complete.
22/11/29 01:24:00 INFO mapred.Task: Using ResourceCalculatorPlugin : null
22/11/29 01:24:00 INFO mapred.LocalJobRunner:
22/11/29 01:24:00 INFO mapred.Merger: Merging 2 sorted segments
22/11/29 01:24:00 INFO mapred.Merger: Down to the last merge-pass, with 2 segments left of total size: 3014 bytes
22/11/29 01:24:00 INFO mapred.LocalJobRunner:
22/11/29 01:24:00 INFO mapred.Task: Task:attempt_local255255534_0001_r_000000_0 is done. And is in the process of committing
22/11/29 01:24:00 INFO mapred.LocalJobRunner:
22/11/29 01:24:00 INFO mapred.Task: Task attempt_local255255534_0001_r_000000_0 is allowed to commit now
22/11/29 01:24:00 INFO output.FileOutputCommitter: Saved output of task 'attempt_local255255534_0001_r_000000_0' to output
22/11/29 01:24:00 INFO mapred.LocalJobRunner: reduce > reduce
22/11/29 01:24:00 INFO mapred.Task: Task 'attempt_local255255534_0001_r_000000_0' done.
22/11/29 01:24:01 INFO mapred.JobClient: map 100% reduce 100%
22/11/29 01:24:01 INFO mapred.JobClient: Job complete: job_local255255534_0001
22/11/29 01:24:01 INFO mapred.JobClient: Counters: 17
22/11/29 01:24:01 INFO mapred.JobClient:   Map-Reduce Framework
22/11/29 01:24:01 INFO mapred.JobClient:     Spilled Records=500
22/11/29 01:24:01 INFO mapred.JobClient:     Map output materialized bytes=3022
22/11/29 01:24:01 INFO mapred.JobClient:     Reduce input records=250
22/11/29 01:24:01 INFO mapred.JobClient:     Map input records=50
22/11/29 01:24:01 INFO mapred.JobClient:     SPLIT_RAW_BYTES=246
22/11/29 01:24:01 INFO mapred.JobClient:     Map output bytes=2510
22/11/29 01:24:01 INFO mapred.JobClient:     Reduce shuffle bytes=0
22/11/29 01:24:01 INFO mapred.JobClient:     Reduce input groups=25
22/11/29 01:24:01 INFO mapred.JobClient:     Combine output records=0
22/11/29 01:24:01 INFO mapred.JobClient:     Reduce output records=25
22/11/29 01:24:01 INFO mapred.JobClient:     Map output records=250
22/11/29 01:24:01 INFO mapred.JobClient:     Combine input records=0
22/11/29 01:24:01 INFO mapred.JobClient:     Total committed heap usage (bytes)=1635778560
22/11/29 01:24:01 INFO mapred.JobClient: File Input Format Counters
22/11/29 01:24:01 INFO mapred.JobClient:   Bytes Read=300
22/11/29 01:24:01 INFO mapred.JobClient: FileSystemCounters
22/11/29 01:24:01 INFO mapred.JobClient:   FILE_BYTES_WRITTEN=162839
22/11/29 01:24:01 INFO mapred.JobClient:   FILE_BYTES_READ=5251
22/11/29 01:24:01 INFO mapred.JobClient: File Output Format Counters
22/11/29 01:24:01 INFO mapred.JobClient:   Bytes Written=196
```

Process finished with exit code 0