Academic year 2019-2020

Department of Computer Science and Engineering

KARNATAKA LAW SOCIETY'S

GOGTE INSTITUTE OF TECHNOLOGY

UDYAMBAG, BELAGAVI-590008

Course Project Report

"Execution of Apache Pig SIZE Function"

Sem: 7

1. Rishi Ladhad 2GI17CS106
2. Shrilakshmi Desai 2GI17CS128
3. Shreya G 2GI17CS124
4. Nidhi Shah 2GI17CS084

Guide
Prof. Arati Shapurkar
Gogte Institute of Technology
Belagavi.

Contents

Definition..... 3
Syntax..... 3
Example..... 3
Verification 5
Program 5
Output..... 6

Definition:

The SIZE() function of Pig Latin is used to compute the number of elements based on any Pig data type.

NOTE –

The return values vary according to the data types in Apache Pig.

Data Type	Value
int, long, float, double	For all these types, the size function returns 1.
Char array	For a char array, the size() function returns the number of characters in the array.
Byte array	For a bytearray, the size() function returns the number of bytes in the array.
Tuple	For a tuple, the size() function returns number of fields in the tuple.
Bag	For a bag, the size() function returns number of tuples in the bag.
Map	For a map, the size() function returns the number of key/value pairs in the map.

Syntax:

Given below is the syntax of the **SIZE()** function.

grunt> SIZE(expression)

Example:

Assume that we have a file named **student.txt** in the HDFS directory **/pig_data/** as shown below.

students.txt

001,Shreya,21,9848022337,Hyderabad,89

002,Nidhi,22,9848022338,Kolkata,78
003,Rishi,22,9848022339,Delhi,90
004,Shrilakshmi,21,9848022330,Pune,93

And the file is loaded into Pig with the relation name **student** as shown below.

```
student = LOAD '/root/Desktop/students' USING PigStorage(',')

as (id:int, firstname:chararray, age:int, phone:chararray, city:chararray, gpa:int);
```

Calculating the Size of the Type

To calculate the size of the type of a particular column, we can use the **SIZE()** function. Let's calculate the size of the name type as shown below.

```
grunt> size = FOREACH student GENERATE SIZE(name);
```

Verification:

Verify the relation **size** using the **DUMP** operator as shown below.

```
grunt > Dump size;
```

Output:

It will produce the as shown, displaying the contents of the relation **size**.

(6)(5)(5)(9)

Program:

```
student_details = LOAD '/root/Desktop/students' USING PigStorage(',')

as (id:int, firstname:chararray, age:int, phone:chararray, city:chararray, gpa:int);

size = FOREACH student_data GENERATE SIZE(name);

dump size;
```

Output:

```
Success!

Job Stats (time in seconds):
JobId   Alias   Feature Outputs
job_local48990663_0013  size,student_details    MAP_ONLY        file:/tmp/temp-675115325/tmp-1321608967,

Input(s):
Successfully read records from: "/root/Desktop/students"

Output(s):
Successfully stored records in: "file:/tmp/temp-675115325/tmp-1321608967"

Job DAG:
job_local48990663_0013


2020-07-21 02:11:13,931 [main] INFO  org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - Success!
2020-07-21 02:11:13,932 [main] WARN  org.apache.pig.data.SchemaTupleBackend - SchemaTupleBackend has already been initialized
2020-07-21 02:11:13,933 [main] INFO  org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1
2020-07-21 02:11:13,933 [main] INFO  org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1
(6)
(5)
(5)
(9)
grunt>
```