RDD Operations in PySpark

Big Data & Predictive Analysis Lanjut

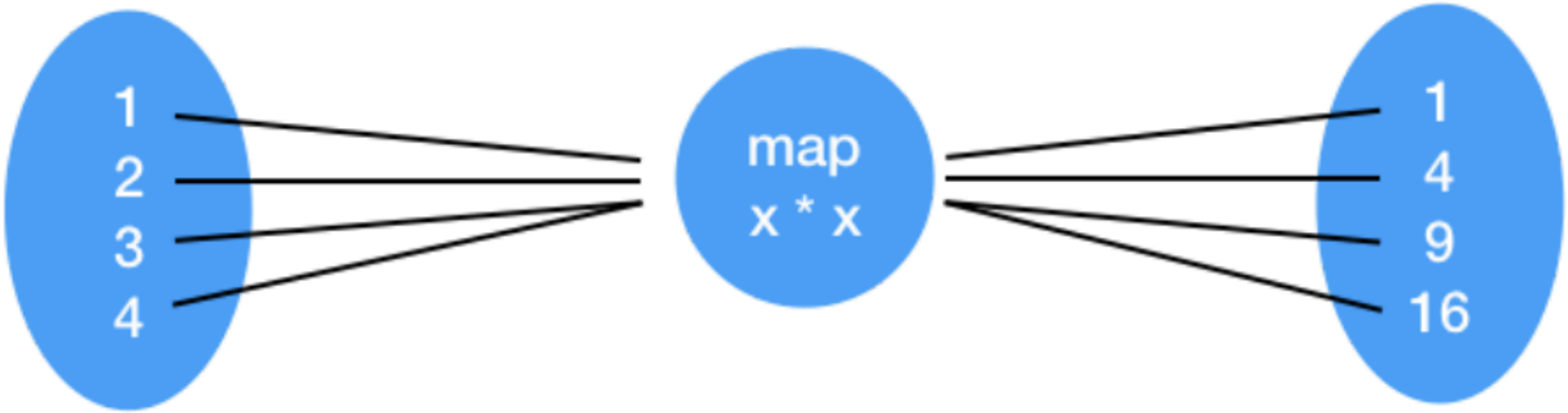
Arif Laksito, M. Kom

Overview of PySpark operations

- Spark operations = Transformations + Action
- Transformations create new RDDs
- Actions perform computation on the RDDs
- Spark operations = Transformations + Action
- Basic RDD Transformations

  `map( )`, `filter( )`, `flatMap( )`, and `union( )`

map( ) transformation

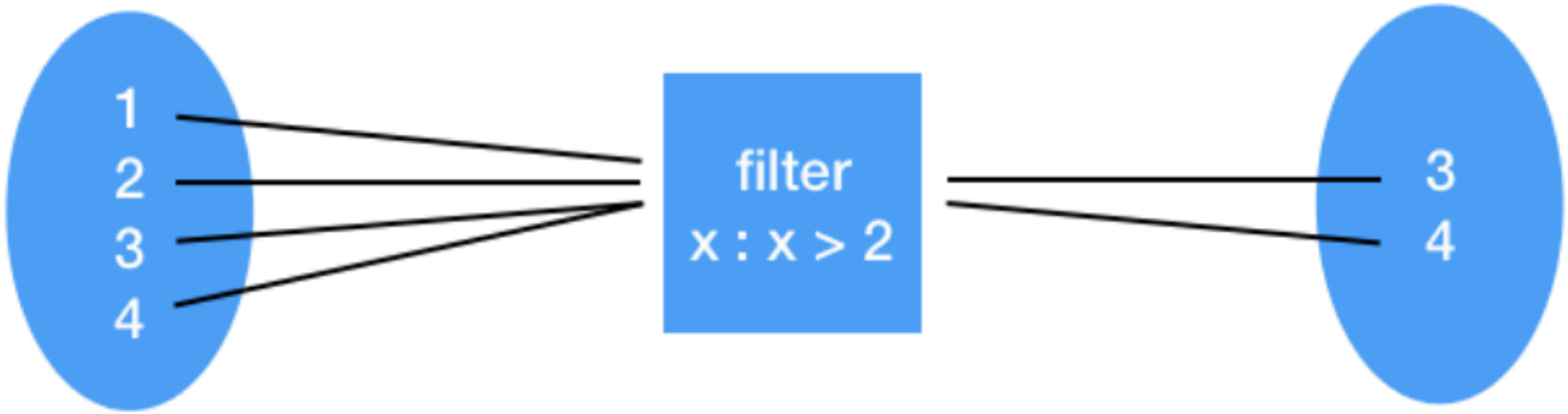


- map( ) transformation applies a function to all elements in the RDD

```
RDD = sc.parallelize([1,2,4,5])

RDD_map = RDD.map(lambda x: x * x)
```

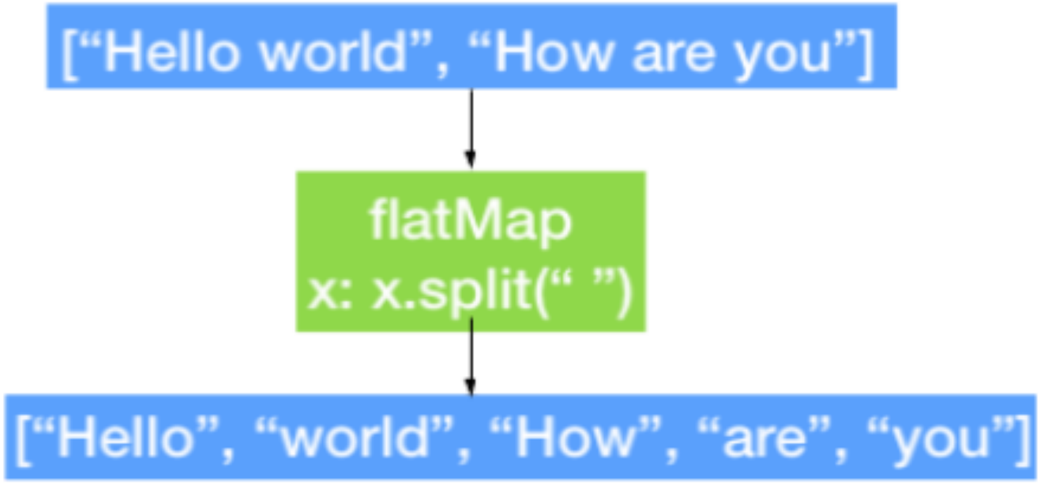
filter( ) transformation



- filter( ) transformation returns a new RDD with only the elements that pass the conditions

```
RDD = sc.parallelize([1,2,4,5]
RDD_map = RDD.filter(lambda x: x > 2)
```

flatMap( ) transformation



- flatMap( ) transformation returns multiple values for each element in the original RDD

```
RDD = sc.parallelize([“hello world”, ”how are you”])
RDD_map = RDD.flatMap(lambda x: x.split(“ “))
```

union( ) transformation


```
inputRDD = sc.textFile("logs.txt")

errorRDD = inputRDD.filter(lambda x:"error" in x.split())

warningsRDD = inputRDD.filter(lambda x:"warning" in x.split())

combinedRDD = errorRDD.union(warningsRDD)
```

RDD Actions

- Operation return a value after running a computation on the RDD
- Basic RDD Actions:
  - `collect( )`
  - `take(N)`
  - `first( )`
  - `count( )`

collect & take Actions

- collect() return all the elements of dataset as an array
- take(N) returns an array with the first N elements of the dataset
  - `RDD_map.collect()`
  - `RDD_mak.take(5)`

Let's code

PySpark Data Manipulation