

Project2 实验报告

一. 数组平方和计算

Spark 基于 Ubuntu 的部署因为作业介绍中已经有了详细的介绍, 这里我们仅通过图 1 即可看出 Spark 已成功部署。

学号后三位 335 ($335\%11 + 20 = 25$), 我们使用 `randint(0,25)`, 运行截图参见图 1

```

monstery@ubuntu: ~/Desktop/spark-2.1.0-bin-hadoop2.7
17/05/02 16:54:14 WARN Utils: Set SPARK_LOCAL_IP if you need to bind to another
address
17/05/02 16:54:21 WARN ObjectStore: Failed to get database global_temp, returnin
g NoSuchObjectException
Welcome to

Spark version 2.1.0

Using Python version 2.7.12 (default, Nov 19 2016 06:48:10)
SparkSession available as 'spark'.
>>> import random
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ImportError: No module named random
>>> import random
>>> array = sc.parallelize([random.randint(0, 25) for i in range(10)])
>>> array.collect()
[18, 1, 8, 22, 1, 21, 6, 29, 4, 11]
>>> array.map(lambda x: x*x).reduce(lambda a, b: a + b)
1888
>>>

rhythm@rhythmCao: ~/spark-2.1.0-bin-hadoop2.7/bin
Welcome to

Spark version 2.1.0

Using Python version 2.7.12 (default, Nov 19 2016 06:48:10)
SparkSession available as 'spark'.
>>> text_file=sc.textFile('../README.md')
>>> counts=text_file.flatMap(lambda line:line.split(' ')).map(lambda word:(word,
1)).reduceByKey(lambda a,b:a+b)
>>> from pprint import pprint
>>> pprint(counts.collect())
[(u'', 72),
 (u'when', 1),
 (u'R', 1),
 (u'including', 4),
 (u'computation', 1),
 (u'contributing', 1),
 (u'submit', 1),
 (u'using', 1),
 (u'guidance', 2),
 (u'Scala', 1),
 (u'environment', 1),
 (u'only', 1),
 (u'rich', 1),
 (u'Apache', 1),
 (u'sc.parallelize(range(1000)).count()', 1),
 (u'Building', 1),
 (u'IDE', 1),
 (u'guide', 1),

```

图 1: 数组平方和

图 2: word count 统计

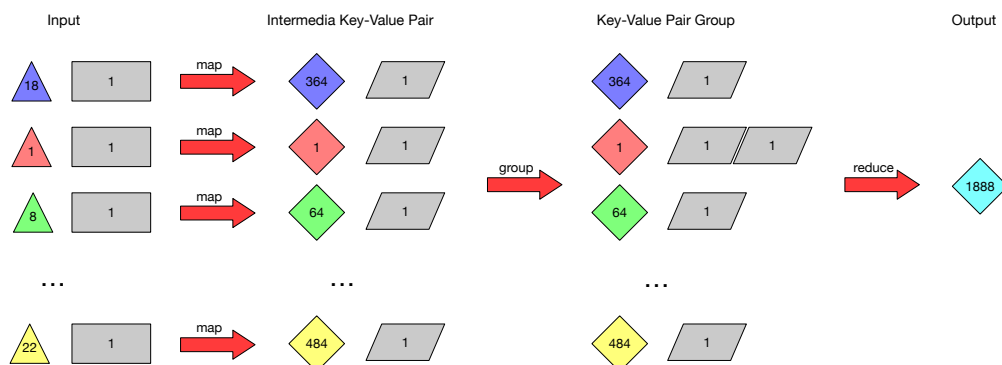


图 3: 数组平方和 MapReduce 示意图

二. word count 统计

运行截图参见图 2。

以 word count 计数为例, 输入文件 `README.md`, 由于该文件缺少代表性, 我们以一个三行的文本为例. 任务分为 map 和 reduce 两个阶段:

- Map 阶段

并行读取文件每一行的输入, 对读取的单词进行 Map 操作, 每个单词都以 <key, value> 的形式生成, 参见图 5.

```
flatMap(lambda line:line.split(' ')).map(lambda word: (word, 1))
```

- Reduce 阶段

对 map 的结果进行排序, 合并, 最后得出词频, 参见图 6.

```
reduceByKey(lambda a, b: a+b)
```

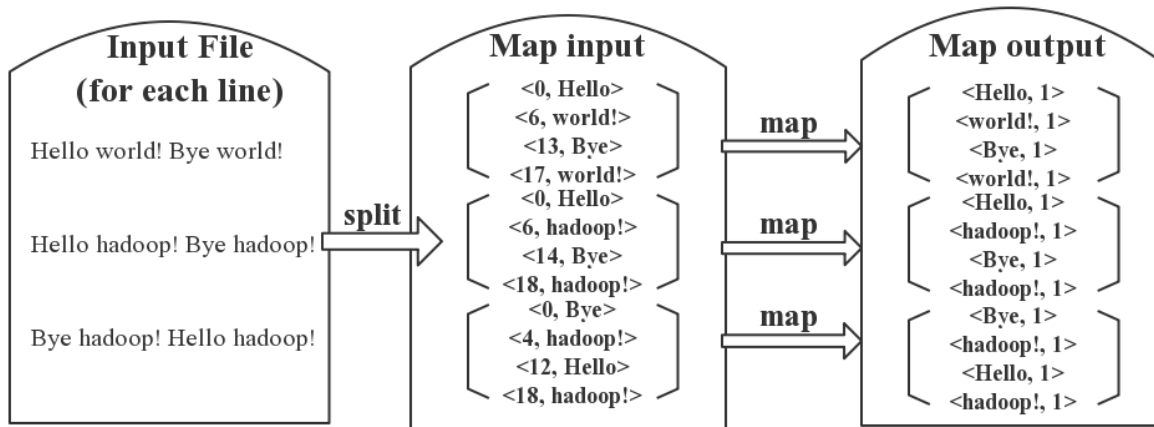


图 5: Map 阶段

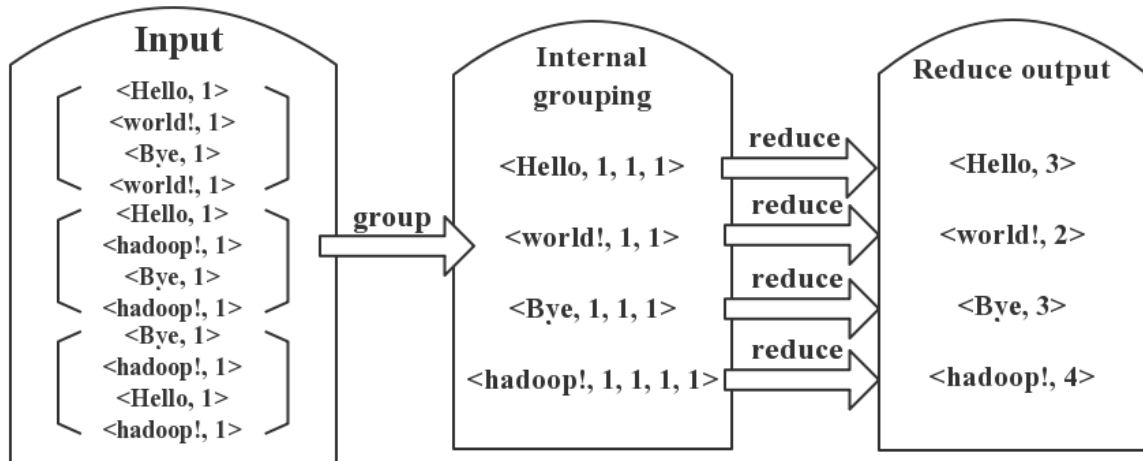


图 6: Reduce 阶段