

CSDN

博客学院下载GitChatTinyMind论坛APP问答商城VIP会员活动招聘ITeye

搜博

原

2

rk2.10中使用累加器、注意点以及实现自定义累加器

2017 24日 19:38:42 阅读数：6367 标签：spark累加器更多

文为博主原创文章，欢迎转载和交流，转载请注明出处^^ https://blog.csdn.net/u013468917/article/details/70617085

累加器 (accumulator) 是Spark中提供的一种分布式的变量机制，其原理类似于mapreduce，即分布式的改变，然后聚合这些改变。累加器的一个常见用途是在分布式计算中统计某些数据的总和。

累加器 用于

Spark 提供了Long和Double类型的累加器。下面是一个简单的使用示例，在这个例子中我们在过滤掉RDD中奇数的同时进行计数，最后计算剩下整数的和。

```
1 val sparkConf = new SparkConf().setAppName("Test").setMaster("local[2]")
2 val sc = new SparkContext(sparkConf)
3 val accum = sc.longAccumulator("longAccum") //统计奇数的个数
4 val sum = sc.parallelize(Array(1,2,3,4,5,6,7,8,9),2).filter(n=>{
5     if(n%2!=0) accum.add(1L)
6     n%2==0
7 }).reduce(_+_ )
8
9 println("sum: "+sum)
10 println("accum: "+accum.value)
11
12 sc.stop()
```

结果为：

sum: 20
accum: 5

这是结果正常的情况，但是在使用累加器的过程中如果对于spark的执行过程理解的不够深入就会遇到两类典型的错误：少加（或者没加）、多加。

少加的情况：

对于如下代码：

```
1 val accum = sc.longAccumulator("longAccum")
2 val numberRDD = sc.parallelize(Array(1,2,3,4,5,6,7,8,9),2).map(n=>{
3     accum.add(1L)
4     n+1
5 })
6 println("accum: "+accum.value)
```

执行完毕，打印的值是多少呢？答案是0，因为累加器不会改变spark的lazy的计算模型，即在打印的时候像map这样的transformation还没有真正的执行，从而累加器的值也没有增加。

多加的情况：

对于如下代码：

```
1 val accum = sc.longAccumulator("longAccum")
2 val numberRDD = sc.parallelize(Array(1,2,3,4,5,6,7,8,9),2).map(n=>{
3     accum.add(1L)
4     n+1
5 })
6 numberRDD.count
7 println("accum1: "+accum.value)
8 numberRDD.reduce(_+_ )
9 println("accum2: "+accum.value)
```



心理学



联系我们



请扫描二维码
webmaster@csdn.net
400-660-0000
QQ客服

关于 招聘 广告服务 网络110报警服务 中国互联网举报中心 北京互联网违法和不良信息举报中心

```
accum1:9
```

```
accum2: 18
```

我们虽然只在map里进行了累加器加1的操作，但是两次得到的累加器的值却不一样，这是由于count和reduce都是action类型的操作，触发了两次作业的提交，所以map算子实际上被执行了两次，在作业后累加器又完成了一轮计数，所以最终累加器的值为18。究其原因是因为count虽然促使numberRDD被计出来，但是由于没有对其进行缓存，所以下次再次需要使用numberRDD这个数据集是，需要的部分开始执行计算。解释到这里，这个问题的解决方法也就很清楚了，就是在count之前调用numberRDD的cache方法（或persist），这样在count后数据集就会被缓存下来，reduce操作就会读取缓存从头开始计算了。改成如下代码即可：

```
1  val accum = sc.longAccumulator("longAccum")
2  val numberRDD = sc.parallelize(Array(1,2,3,4,5,6,7,8,9),2).map(n=>{
3      accum.add(1L)
4      n+1
5  })
6  numberRDD.cache().count
7  println("accum1:"+accum.value)
   numberRDD.reduce(_+_ )
   println("accum2: "+accum.value)
```

这次两次打印的值就会保持一致了。

自定义累加器

自定义累加器类型的功能在1.X版本中就已经提供了，但是使用起来比较麻烦，在2.0版本后，累加器的易用性有了较大的改进，而且官方还提供了一个新的抽象类：AccumulatorV2来提供更加友好累加器的实现方式。官方同时给出了一个实现的示例：CollectionAccumulator类，这个类允许以集合的形式收集spark应用执行过程中的一些信息。例如，我们可以用这个类收集Spark一些细节，当然，由于累加器的值最终要汇聚到driver端，为了避免 driver端的outofmemory问题，需要对收集的信息的规模要加以控制，不宜过大。

实现自定义类型累加器需要继承AccumulatorV2并至少覆写下例中出现的方法，下面这个累加器可以用于在程序运行过程中收集一些文本类信息，最终以Set[String]的形式返回。

```
1  import java.util
2
3  import org.apache.spark.util.AccumulatorV2
4
5  class LogAccumulator extends AccumulatorV2[String, java.util.Set[String]] {
6      private val _logArray: java.util.Set[String] = new java.util.HashSet[String]()
7
8      override def isZero: Boolean = {
9          _logArray.isEmpty
10     }
11
12     override def reset(): Unit = {
13         _logArray.clear()
14     }
15
16     override def add(v: String): Unit = {
17         _logArray.add(v)
18     }
19
20     override def merge(other: AccumulatorV2[String, java.util.Set[String]]): Unit = {
21         other match {
22             case o: LogAccumulator => _logArray.addAll(o.value)
23         }
24     }
25
26
27     override def value: java.util.Set[String] = {
28         java.util.Collections.unmodifiableSet(_logArray)
29     }
30
31     override def copy(): AccumulatorV2[String, util.Set[String]] = {
32         val newAcc = new LogAccumulator()
33         _logArray.synchronized{
34             newAcc._logArray.addAll(_logArray)
35         }
36         newAcc
37     }
38 }
```

测试类：

```
1 import scala.collection.JavaConversions._
2
3 import org.apache.spark.{SparkConf, SparkContext}
4
5 object Main {
6   def main(args: Array[String]): Unit = {
7     val sparkConf = new SparkConf().setAppName("Test").setMaster("local[2]")
8     val sc = new SparkContext(sparkConf)
9     val accum = new LogAccumulator
10    sc.register(accum, "logAccum")
11    val sum = sc.parallelize(Array("1", "2a", "3", "4b", "5", "6", "7cd", "8", "9"), 2).filter(line => {
12      val pattern = """"^-(\d+)""""
13      val flag = line.matches(pattern)
14      if (!flag) {
15        accum.add(line)
16      }
17      flag
18    }).map(_._2.toInt).reduce(_ + _)
19
20    println("sum: " + sum)
21    for (v <- accum.value) print(v + " ")
22    println()
23    sc.stop()
24  }
25 }
```

本例中利用自定义的收集器收集过滤操作中中被过滤掉的元素，当然这部分的元素的数据量不能太大。运行结果如下：

sum; 32

7cd 4b 2a

Bug报告 (3.29-23:26)

undefined

想对作者说点什么？

我来说一句

spark自定义Accumulator高级应用 (JAVA)

 1853

public class SessionAggrStatAccumulator implements AccumulatorParam { private static final long ...

spark2.1.0自定义累加器AccumulatorV2的使用

 3937

spark2.1.0自定义累加器AccumulatorV2的使用

Spark 2.x 自定义累加器AccumulatorV2的使用 - CSDN博客

废除Spark2.x之后,之前的的accumulator被废除,用AccumulatorV2代替;更新增加创建并注册一个long accumulat...

Spark 2.X 自定义AccumulatorV2 JavaAPI实现 - CSDN博客

自定义Accumulator:Scala自定义accumulator代码:import org.apache.spark.util.AccumulatorV2 class MyAccum...



短信验证码平台

百度广告

spark之共享数据(累加器)

 148

累加器顾名思义，累加器是一种只能通过关联操作进行“加”操作的变量，因此它能够高效的应用于并行操作中。...

spark2.1.0自定义累加器AccumulatorV2的使用 - CSDN博客

isZero: 当AccumulatorV2中存在类似数据不存在这种问题时,是否结束程序。 copy:...spark自定义Accumulator高...

Spark Accumulator的正确使用方式 - CSDN博客

JavaSparkContext sc = new JavaSparkContext(conf);SQLContext sqlContext = ...spark2.1.0自定义累加器Accu...

JavaSpark-编程进阶-累加器

489

spark的一些进阶特性 累加器（accumulate）：用于聚合和统计 广播变量（broadcast variable）：高效分发...

RDD 累加器

64

累加器累加器用来对信息进行聚合，通常在向 Spark 传递函数时，比如使用map() 函数或者用 filter() 传条件时...

spark自定义Accumulator高级应用(JAVA) - CSDN博客

isEmpty(v1)) { return v2; } // 使用StringUtils工具类,从v1中,提取v2...JavaSparkContext sc = new JavaSparkCont...

第39课:Spark中的Broadcast和Accumulator机制解密 - CSDN博客

1,自定义的时候可以让Accumulator非常复杂,基本上可以是任意类型的Java和Scala对象...Accumulator已经被标...

源码阅读-累加器(十)

177

冯阅读-累加器(十) 使用场景 累加器是一种支持并行只能added的特殊变量,常用来计次/求和,我们也...

成都小两口下班没事在家赚钱，半年后存款惊人！

翔灿咨询 · 顶新

spark 2.2.0 accumulator使用方法 java版 python版 - CSDN博客

java版 package cn.spark.study.core; import org.apache.spark.Accumulator; import org.apache.spark.SparkCon...

Spark2.X 使用累加器AccumulatorV2实现字符串拼接下的..._CSDN博客

Spark 2.X 中的累加器和 Spark 1.X中有着很大不同,下面将实现的功能是:将一个集合,集合中含有字母 "A","B","A"...

Spark累加器(Accumulator)陷阱及解决办法

1.3万

程序中可能会使用到spark提供的累加器功能，可是如果你不了解它的运行机制，有时候会带来一些负面作用（ ...

Spark自定义累加器

447

package com.sparkproject.abc; import org.apache.spark.AccumulatorParam; public class UDFAccumulat...

Spark累加器(Accumulator)使用详解 - CSDN博客

Spark 2.x 自定义累加器AccumulatorV2的使用 废除Spark2.x之后,之前的的accumulator...java并发编程 阅读量:3...

spark2.x-Accumulator - CSDN博客

import java.util import org.apache.spark.util.AccumulatorV2 class LogAccumulator extends AccumulatorV2[Stri...

Spark累加器使用

7622

Spark累加器使用 使用spark累加器，解决视频平均播放数计算，以及视频播放数平方和平均值 val totalTimes=s...

Spark自定义累加器的实现

5398

1.为什么要使用自定义累加器前文讲解过spark累加器的简单使用：http://blog.csdn.net/lxhandlbb/article/details/...

文章热词

spark中的sc是什么意思 spark修改临时目录 spark 写orc文件 sparksql 按时间排序 sparkml中的als算法



verilog实现的累加器程序

2010年03月03日 635B 下载



spark中用scala编写累加器小程序统计文章中空白行

2017年03月06日 682B 下载



quartus ii 四位累加器原理图工程

2012年12月19日 396KB 下载

老中医说：男人多吃这个东西，时间延长5倍！

番当生物 · 顶新



基于FPGA的乘累加器

2011年12月22日 5KB 下载



VHDL的累加器

2014年10月27日 75KB 下载

Spark的广播和累加器的使用

1.2万

广播和计数器的解释1.1 广播：广播变量允许程序员将一个只读的变量缓存在每台机器上，而不用在任务之间传...

spark广播变量和累加器

1790

spark广播变量和累加器 广播变量 Spark中分布式执行的代码需要传递到各个Executor的Task上运行。对于一些...

Spark累加器(Accumulator)使用详解

313

accumulator[T](initialValue: T,name: String)(implicit param: org.apache.spark.AccumulatorPar...



一点点加盟费

百度广告



FPGA 累加器程序

2009年08月09日 227KB 下载



经典Hough算法的实现

2018年04月25日 158KB 下载

个人资料



古月慕南

原创
68

粉丝
56

等级： 博客 5
积分： 2499
勋章： 恒

访问
排名



一点点加盟费



最新文章

spring与mybatis整合
java.lang.NoSuchMethodError
思路
对spark2.3.0中Structured
持续处理模式的介绍

CDH 5.12.1 离线部署指南
成指南)
git开发流程总结

个人分类

编程-器

BigData-器

BigData-术

编程-术

javaFF-SSM

展开

归档

2018年6月

2018年5月

2018年3月

2017年10月

2017年7月

展开

热门文章

spark将数据写入hbase以
据
阅读量：40236

Spring-jdbc：JdbcTempla
阅读量：39279

在java应用中使用JDBC连
iveServer2)
阅读量：10566

ArrayList的contains方法和
nsKey效率差十倍
阅读量：9252

Spring-8：SpEL入门
阅读量：6884

最新评论

Spring-jdbc：JdbcT...
cheercp：很明了，看着舒服。

Spring-jdbc：JdbcT...
wxr15732623310：总结地很不

CDH 5.12.1 离线部署指南
u013468917：[reply]u014134
k的问题，你可以百度一下这个

CDH 5.12.1 离线部署指南
u014134828：你好 我又重装
装spark的时候出现 “仅完成 0/
败：主机 ...

spark将数据写入hbase以
u013468917：[reply]Yoga_L1
你是指？