**Spark Streaming的分析与应用**

**测试需求规格说明书**

Version 1.0.1

小组成员：

陈少杰

姜鑫

蒲彦均

邹嘉欣

**版本变更记录**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本 | 变更时间 | 修改人 | 审核人 | 备注 |
| 1.0.1 | 2017/05/17 | 邹嘉欣、陈少杰、姜鑫、蒲彦均 | 邹嘉欣、陈少杰、姜鑫、蒲彦均 | 初稿 |

# 前言

## 目的

本文档主要用于分析本软件工程综合实验小组对Spark Streaming的分析与应用进行测试的主要内容以及各小组成员之间的分工，在此基础上对测试用例进行了初步的设计。

## 软件测试的主要内容

软件测试是一种实际输出与预期输出间的审核或者比较过程。软件测试的经典定义是：在规定的条件下对程序进行操作，以发现程序错误，衡量软件质量，并对其是否能满足设计要求进行评估的过程。本次软件测试阶段的主要工作如下：

* 总结项目实现的内容，结合需求分析阶段的内容，分析测试需求，编写测试计划及测试规格说明书；
* 编写有效的、覆盖面广的测试用例；
* 研究相关测试技术；
* 按计划实施测试工作，提交测试报告；

## 文档概述

本次测试需求规格说明书主要参照《Spark Streaming的分析与应用-需求规格说明书》以及已经实现的项目内容，给出了需求用例与测试用例的对照表，对于每个测试用例，先给出测试策略的描述，然后按照RUCM4test标准说明测试用例。主要按照功能需求、非功能需求两个方面进行测试。

## 测试用例与需求用例参照表

本次测试严格遵照《Spark Streaming的分析与应用-需求规格说明书》以及项目实现内容来进行设计与实现，测试用例与需求用例的对照表如下：

|  |  |  |
| --- | --- | --- |
| 模块 | 需求用例 | 测试用例 |
| 功能需求 | 数据的产生与输入 | Kafka的发送与接收测试 |
| 数据流抽象DStream | Spark Streaming的数据流构建测试 |
| 长时间容错 | 各个模块的长时间容错测试 |
| 作业调度 | 作业调度测试 |
| 窗口支持 | 窗口支持测试 |
| 非功能需求 | 实时性 | 实时性测试 |
| 扩展性与吞吐量 | Spark Streaming的可扩展性测试；  Spark Streaming的吞吐量测试 |
| 持久化 | Spark Streaming的持久化测试 |
| 性能调优 | 性能调优测试 |

# 2功能需求模块

## 2.1 Kafka的发送与接收测试

### 2.1.1测试策略描述

测试员对Spark Streaming 的一种常见输入源Kafka进行简单的发送与接收测试，将一段数据发送至Kafka的指定topic，并读取此topic，查看输入输出的正确性。

### 2.1.2测试用例

|  |  |  |
| --- | --- | --- |
| **Test Case Specification** | | |
| Name | Kafka的发送与接收测试 | |
| Brief Description | 发送数据后启动数据接收，对照输入输出，检查正确性 | |
| Precondition | Kafka程序正常运行 | |
| Tester | 测试员 | |
| Dependency | None | |
| Test Setup | Name | 准备待发送微博文本文件 |
| Description | 准备待发送微博文本文件 |
| Basic Flow  (Test Setup) | Steps | |
| 1 | 准备待发送微博文本文件 |
| 2 | 检查Kafka是否处于运行状态 |
| Postcondition | Kafka功能正常，数据准备完毕 |
| Basic Flow  (Test Sequence) | Steps | |
| 1 | 将微博文本上传服务器 |
| 2 | 启动Kafka数据发送程序，设置发送数据批数 |
| 3 | 查看发送数据耗时 |
| 4 | 启动接收数据程序 |
| 5 | 查看接收到的数据数量、顺序、耗时等 |
| Postcondition | Kafka发送与接收基本功能正常 |

## 2.2 Spark Streaming的数据流构建测试

### 2.2.1测试策略描述

用例的目的是测试系统是否正确部署并正确启动，脚本分别测试正常启动和异常启动两种情况。正常情况下，系统正确运行并可以进行检索；异常情况下，将出现启动异常或检索异常。

### 2.2.2测试用例

|  |  |  |
| --- | --- | --- |
| **Test Case Specification** | | |
| Name | Spark Streaming的数据流构建测试 | |
| Brief Description | 从Kafka读取数据，构建DStream输入流，直接写入HDFS | |
| Precondition | Kafka、Spark均正常运行 | |
| Tester | 测试员 | |
| Dependency | None | |
| Test Setup | Name | 准备待发送微博文本文件 |
| Description | 准备待发送微博文本文件 |
| Basic Flow  (Test Setup) | Steps | |
| 1 | 准备待发送微博文本文件 |
| 2 | 检查Kafka是否处于运行状态 |
| 3 | 检查Spark是否处于运行状态 |
| Postcondition | Kafka和Spark功能正常，数据准备完毕 |
| Basic Flow  (Test Sequence) | Steps | |
| 1 | 启动Kafka数据发送程序，设置发送数据批数 |
| 2 | 启动Spark测试程序 |
| 3 | 等待数据处理完毕，下载查看HDFS文件 |
| Postcondition | Spark Streaming数据流构建正常 |

## 2.3 Spark Streaming的长时间容错测试

### 2.3.1测试策略描述

用例的目的是测试系统的各个模块在长时间运行状态下对发生的错误情况的应对能力，分别对数据导入导出模块、数据抽象和作业调度模块测试容错能力。正常情况下，当这三个模块出现错误时，系统可以处理错误并继续正常的运行；异常情况下，模块会出现问题无法继续正常运行。

### 2.3.2测试用例

|  |  |  |
| --- | --- | --- |
| **Test Case Specification** | | |
| Name | Spark Streaming的长时间容错测试 | |
| Brief Description | 系统正常运行了一段时间，并且已经完成过一次完整的运行过程 | |
| Precondition | 三个模块均继续正常运行 | |
| Tester | 测试员 | |
| Dependency | None | |
| Test Setup | Name | 为各个模块准备错误数据 |
| Description | 为各个模块准备错误的数据 |
| Basic Flow  (Test Setup) | Steps | |
| 1 | 将三个模块的前置模块的数据保存 |
| 2 | 将保存的数据修改为错误的数据 |
| 3 | 将错误的数据输入到后置模块 |
| Postcondition | 三个模块均接收到错误的数据 |
| Basic Flow  (Test Sequence) | Steps | |
| 1 | 继续运行三个模块 |
| 2 | 检测三个模块的操作内容 |
| 3 | 检测三个模块的运行状态 |
| Postcondition | 三个模块均继续正常运行 |

## 2.4作业调度测试

### 2.4.1测试策略描述

用例的目的是测试系统是否正确地进行作业调度，脚本分别测试作业调度正常和作业调度异常两种情况。正常情况下，系统可以运行相关资源并进行调度；异常情况下，将出现系统异常。

### 2.4.2测试用例

|  |  |  |
| --- | --- | --- |
| **Test Case Specification** | | |
| Name | 作业调度测试 | |
| Brief Description | 测试系统是否正确地进行作业调度 | |
| Precondition | 完成了数据流的抽象 | |
| Tester | 测试员 | |
| Dependency | None | |
| Test Setup | Name | 运行spark streaming系统 |
| Description | 运行spark streaming系统，准备启动任务 |
| Basic Flow  (Test Setup) | Steps | |
| 1 | 输入测试数据； |
| 2 | 系统验证输入数据的完整性； |
| 3 | 将输入的数据流进行离散化； |
| 4 | 将离散化的数据按时间序列划分为RDD； |
| Postcondition  (Test Oracle) | 完成数据流的抽象； |
| Basic Flow  (Test Sequence) | Steps | |
| 1 | 将任务分隔成多个阶段； |
| 2 | 将分隔的作业放入作业调度队列； |
| 3 | 系统检测系统是否有空闲的资源； |
| 4 | 开始后续任务的第一个阶段 |
| Postcondition  (Test Oracle) | 完成任务的执行； |
| Specific Alternative Flows  (Test Sequence) | RFS 3 | |
| 1 | 队列中正在执行的作业占用了所有的资源； |
| 2 | 继续在队列中等待； |
| Postcondition  (Test Sequence) | 系统的队列继续监测正在执行的任务的资源占用情况； |

## 2.5窗口支持测试

### 2.5.1测试策略描述

用例的目的是测试spark streaming应用能否支持不同的窗口大小，在数据入库后，设置多种不同大小的窗口对应用程序的数据进行操作，查看其正确性。

### 2.5.2测试用例

|  |  |  |
| --- | --- | --- |
| **Test Case Specification** | | |
| Name | 窗口支持测试 | |
| Brief Description | 测试能否以不同大小的窗口对数据进行操作； | |
| Precondition | Kafka、spark可以正常运行 | |
| Tester | 测试员 | |
| Dependency | None | |
| Test Setup | Name | 测试数据完全入库，设置窗口大小 |
| Description | 将要进行窗口化操作的数据完全处理存入数据库，并设置多个大小的测试窗口 |
| Basic Flow  (Test Setup) | Steps | |
| 1 | 准备测试数据 |
| 2 | 启动kafka和spark streaming，进行数据处理和入库 |
| 3 | 选定要测试的窗口大小 |
| Postcondition | 测试数据准备完成，各个窗口大小已选定； |
| Basic Flow  (Test Sequence) | Steps | |
| 1 | 选择一个大小的窗口； |
| 2 | 窗口在数据块上滑动； |
| 3 | 合并窗口内的数据块； |
| 4 | 判断合并后的数据块是否正确 |
| 5 | 循环执行1-4，直至各个选定大小的窗口测试完毕 |
| Postcondition | 窗口支持功能是否正常； |

# 3非功能需求模块

**3.1 实时性测试**

**3.1.1测试策略描述**

用例的目的是spark streaming应用能否满足实时性的要求，即测试系统输入数据后，到处理完成所需的时间。设置多个批次的微博数据，对每批次数据发送到处理完成的用时，进而判断其实时性是否满足要求。

**3.1.2测试用例**

|  |  |  |
| --- | --- | --- |
| **Test Case Specification** | | |
| Name | 实时性测试 | |
| Brief Description | 测试数据输入到数据处理的时间； | |
| Precondition | Kafka和spark运行正常 | |
| Tester | 测试员 | |
| Dependency | None | |
| Test Setup | Name | 准备待发送的文本 |
| Description | 准备待发送微博文本文件； |
| Basic Flow  (Test Setup) | Steps | |
| 1 | 准备待发送微博文本文件； |
| 2 | 确保kafka和spark处于运行状态； |
| Postcondition | Kafka和Spark功能正常，数据准备完毕； |
| Basic Flow  (Test Sequence) | Steps | |
| 1 | 启动kafka数据发送程序，设置发送数据批数； |
| 2 | 启动spark测试程序； |
| 3 | 每批次数据处理完毕，记录该批次处理时间； |
| 4 | 计算批次平均处理时间 |
| Postcondition | 实时性是否符合实际要求； |

## 3.2 Spark Streaming的可扩展性测试

### 3.2.1测试策略描述

在集群现有规模下再添加一台机器，做可扩展性测试。

### 3.2.2测试用例

|  |  |  |
| --- | --- | --- |
| **Test Case Specification** | | |
| Name | Spark Streaming的可扩展性测试 | |
| Brief Description | 向Spark集群中新增一台机器，经少量配置，完成集群扩展，达到可用 | |
| Precondition | Spark集群正常运行 | |
| Tester | 测试员 | |
| Dependency | None | |
| Test Setup | Name | 确认Spark集群运行情况 |
| Description | 确认Spark集群运行情况 |
| Basic Flow  (Test Setup) | Steps | |
| 1 | 对新加入的节点进行安装配置 |
| 2 | 修改现有集群的配置 |
| 3 | 重新启动新集群 |
| Postcondition | 新集群运行正常 |
| Basic Flow  (Test Sequence) | Steps | |
| 1 | 启动Kafka数据发送程序，设置发送数据批数 |
| 2 | 启动Spark数据接收测试程序 |
| 3 | 查看Spark Application管理页面，检查新节点是否成功运行 |
| Postcondition | 整个集群，包括新节点，运行正常 |

## 3.3 Spark Streaming的吞吐量测试

### 3.3.1测试策略描述

在集群正常运行状态下对输入数据直接进行输出，从而统计系统吞吐量。

### 3.3.2测试用例

|  |  |  |
| --- | --- | --- |
| **Test Case Specification** | | |
| Name | Spark Streaming的吞吐量测试 | |
| Brief Description | 在集群正常运行状态下对输入数据直接进行输出，从而统计系统吞吐量 | |
| Precondition | Spark集群正常运行 | |
| Tester | 测试员 | |
| Dependency | None | |
| Test Setup | Name | 确认Spark集群、Kafka运行情况 |
| Description | 确认Spark、Kafka集群运行情况 |
| Basic Flow  (Test Setup) | Steps | |
| 1 | 准备测试用数据 |
| 2 | Kafka发送程序部署到多台机器 |
| Postcondition | 保证发送速度不低于Spark Streaming处理速度 |
| Basic Flow  (Test Sequence) | Steps | |
| 1 | 启动Kafka发送数据程序，以最快速度发送数据 |
| 2 | 启动Spark Streaming简单读写程序 |
| 3 | 查看运行日志，记录运行时间与处理数据量的关系 |
| Postcondition | 根据实际测试情况得出集群吞吐量 |

## 3.4 Spark Streaming的持久化测试

### 3.4.1测试策略描述

用例的目的是测试系统把临时数据保存到可多次读取的设备上的能力。主要测试少量的临时数据持久化到内存的能力和大量数控持久化到磁盘的能力。

### 3.4.2测试用例

|  |  |  |
| --- | --- | --- |
| **Test Case Specification** | | |
| Name | Spark Streaming的持久化测试 | |
| Brief Description | 系统运行正常 | |
| Precondition | 数据持久化成功 | |
| Tester | 测试员 | |
| Dependency | None | |
| Test Setup | Name | 测试数据的持久化 |
| Description | 测试数据的持久化 |
| Basic Flow  (Test Setup) | Steps | |
| 1 | 测试系统生成了一些大小不同的临时数据 |
| 2 | 将不同大小的数据依次传入系统中 |
| 3 | 系统开启持久化操作 |
| Postcondition | 系统接收到了临时数据 |
| Basic Flow  (Test Sequence) | Steps | |
| 1 | 检测临时数据的大小情况和读取需求 |
| 2 | 选择不同的持久化策略，针对小的数据进行内存持久化，大的数据进行磁盘持久化 |
| 3 | 查看内存或者磁盘中是否有持久化数据 |
| Postcondition | 临时数据成功持久化 |

**3.4 性能调优测试**

**3.4.1测试策略描述**

用例的目的是测试系统的性能调优方面的能力，主要测试批处理时间、并行化程度和过期数据清除三部分测试。批处理时间主要测试每批次数据处理时间与批处理时间间隔设置是否合理，并行化程度主要测试kafka发送端的并行发送能力，清除过期数据主要测试过期数据删除时间是否合理。

**1.4.2测试用例**

|  |  |  |
| --- | --- | --- |
| **Test Case Specification** | | |
| Name | 批处理时间测试 | |
| Brief Description | 测试批处理时间是否合理； | |
| Precondition | Kafka和spark运行正常 | |
| Tester | 测试员 | |
| Dependency | None | |
| Test Setup | Name | 准备待发送的文本 |
| Description | 准备待发送微博文本文件； |
| Basic Flow  (Test Setup) | Steps | |
| 1 | 准备待发送微博文本文件； |
| 2 | 确保kafka和spark处于运行状态； |
| Postcondition | Kafka和Spark功能正常，数据准备完毕； |
| Basic Flow  (Test Sequence) | Steps | |
| 1 | 启动kafka数据发送程序； |
| 2 | 启动spark测试程序； |
| 3 | 每批次数据处理完毕，记录该批次数据处理时间、等待处理时间； |
| 4 | 计算批次平均处理时间、平均等待处理时间 |
| 5 | 判断数据处理时间和等待处理时间的壁纸是否合适 |
| Postcondition | 判断当前批处理时间是否合适； |

|  |  |  |
| --- | --- | --- |
| **Test Case Specification** | | |
| Name | 并行化程度测试 | |
| Brief Description | 测试kafka获取数据的并行能力； | |
| Precondition | Kafka和spark运行正常 | |
| Tester | 测试员 | |
| Dependency | None | |
| Test Setup | Name | 准备待发送的文本，设置多个kafka实例 |
| Description | 准备待发送微博文本文件，设置多个kafka实例； |
| Basic Flow  (Test Setup) | Steps | |
| 1 | 准备待发送微博文本文件； |
| 2 | 确保kafka和spark处于运行状态； |
| Postcondition | Kafka和Spark功能正常，数据准备完毕； |
| Basic Flow  (Test Sequence) | Steps | |
| 1 | 启动kafka数据发送程序； |
| 2 | 启动spark测试程序； |
| 3 | 测试并行化后个kafka实例的数据获取能力； |
| 4 | 判断kafka数据获取部分能否很好实现并行化 |
| Postcondition | 判断kafka是否有很好的并行化处理能力； |

|  |  |  |
| --- | --- | --- |
| **Test Case Specification** | | |
| Name | 过期数据清理测试 | |
| Brief Description | 测试过期数据能否及时清除 | |
| Precondition | Kafka和spark运行正常 | |
| Tester | 测试员 | |
| Dependency | None | |
| Test Setup | Name | 数据不停入库，设置删除时间 |
| Description | Kafka不停地获取微博数据，并设置要删除数据的时间； |
| Basic Flow  (Test Setup) | Steps | |
| 1 | 启动kafka，不停获取数据 |
| 2 | 设置定期删除时间（为了测试方便，可设置为1h） |
| Postcondition | Kafka正常运行，设置了数据清除时间； |
| Basic Flow  (Test Sequence) | Steps | |
| 1 | 启动kafka，不停获取数据 |
| 2 | 设置定期删除时间（为了测试方便，可设置为1h） |
| 3 | 每隔一段时间记录过期数据的删除情况 |
| Postcondition | 确保过期数据能及时清理； |