第 8 章 数据分析—电商 PV、UV 榜实战

★本章导读★

本章通过电商领域常见的业务需求，如 PV、UV 统计，将前面章节所涉及的 RDD 相关算子进行练习。通过实际业务的开发使读者更深层次理解 RDD 算子的适用场景，能够熟练的将 RDD 算子应用。进一步，通过实战练习，能够初步培养面对业务需求，提炼处理问题方法的思路。

★知识要点★

通过本章内容的学习，读者将掌握以下知识：

- RDD 常见算子的应用
- 电商 PV、UV 指标含义
- 能熟练适用 RDD 进行数据分析处理

8.1 数据统计实战介绍

电商行业是大数据分析、处理应用最多的场景之一。在电商场景中，我们经常听到相关的指标名称，例如：日活、点击率、曝光率、转换率等等。此类指标都和大数据分析处理息息相关，本章节将通过常见的两个指标的实战，巩固前一章节的学习。

8.1.1 电商 PV、UV 需求介绍

什么是 PV？

PV（Page View）即页面浏览量或点击量，是网站分析的一个术语，是衡量一个网站或网页用户访问量的指标。具体的说，PV 值就是所有访问者在 24 小时（0 点到 24 点）内看了某个网站多少个页面或某个网页多少次。比如，淘宝网某日 PV 为 123 万，PV 是指页面刷新的次数，每一次页面刷新，就算做一次 PV 流量。对于广告主，PV 值可预期它可以带来多少广告收入。一般来说，PV 与来访者的数量成正比，但是 PV 并不直接决定页面的真实来访者数量，如同一个来访者通过不断的刷新页面，也可以制造出非常高的 PV。

PV 值的度量方法就是从浏览器发出一个对网络服务器的请求（Request），网络服务器接到这个请求后，会将该请求对应的一个网页（Page）发送给浏览器，从而产生了一个 PV。那么在这里只要是这个请求发送给了浏览器，无论这个页面是否完全打开（下载完成），那么都是应当计为 1 个 PV。

什么是 UV？

UV（Unique Visitor）即独立访客数，指访问某个站点或点击某个网页的不同 IP 地址的人数。在同一天内，UV 只记录第一次进入网站的具有独立 IP 的访问者，在同一天内再次访问该网站则不计数，即一天内同个访客多次访问仅计算一个 UV。UV 提供了一定时间内不同观众数量的统计指标，而没有反应出网站的全面活动。在电商领域中，UV 价值越大，产品越迎合消费者需求，只有一定的推广投入才会带来相对应的 UV

通过统计并分析每日网站的 PV、UV 值，可以帮助企业很好的把控当前业务或者广告投入的效益

情况，对电商产品经理、运营做出合理的战略调整提供有力支持。

8.1.2 需求分析及思路

本章节将对几大网站某日的用户访问日志数据进行处理分析，计算各个电商网站 PV、UV 值，并按 PV、UV 排序输出，了解各个网站的流量情况。需求思路如下：

1. 读取日志数据，查看日志数据结构；
2. 解析日志数据，分组统计各个网站 PV、UV 值；
3. 分别按 PV、UV 值降序排序输出。

需求实现思路比较简单，核心问题在于对日志数据的解析处理和数据去重方法（计算 UV 指标）。

8.2 电商 PV、UV 榜代码实战

本小节中，我们将介绍如何从读取用户行为访问日志数据开始，一步一步实现电商网站 PV、UV 的统计计算。

8.2.1 数据集介绍

本章节实战使用的数据为各个电商网站某日用户访问数据，数据文件为 access_log.txt，数据内容如下图 8-1 所示。

```
105.145.171.151→辽宁——→2021-06-01→1532057592615——→4389195290417379165→www.jd.com→View
105.145.171.151→辽宁——→2021-06-01→1532057592616——→4389195290417379165→www.dangdang.com——→Login
105.145.171.151→辽宁——→2021-06-01→1532057592617——→4389195290417379165→www.taobao.com→Comment
156.179.160.30→吉林——→2021-06-01→1532057592617——→5325897281591868217→www.suning.com→Regist
156.179.160.30→吉林——→2021-06-01→1532057592617——→5325897281591868217→www.dangdang.com——→Comment
156.179.160.30→吉林——→2021-06-01→1532057592618——→5325897281591868217→www.baidu.com→View
190.37.92.153→北京——→2021-06-01→1532057592618——→5442936660226459177→www.suning.com→Click
190.37.92.153→北京——→2021-06-01→1532057592619——→5442936660226459177→www.dangdang.com——→Buy
190.37.92.153→北京——→2021-06-01→1532057592619——→5442936660226459177→www.gome.com.cn→Regist
190.37.92.153→北京——→2021-06-01→1532057592619——→5442936660226459177→www.dangdang.com——→Buy
190.37.92.153→北京——→2021-06-01→1532057592620——→5442936660226459177→www.dangdang.com——→Buy
14.192.111.162→浙江——→2021-06-01→1532057592620——→2729036904202283535→www.baidu.com→Click
157.140.205.43→山西——→2021-06-01→1532057592620——→7525107213152810709→www.suning.com→Comment
94.0.201.117→湖南——→2021-06-01→1532057592621——→4926659864297255839→www.gome.com.cn→Click
161.59.30.196→山西——→2021-06-01→1532057592621——→3572062651028442022→www.baidu.com→Click
161.59.30.196→山西——→2021-06-01→1532057592621——→3572062651028442022→www.baidu.com→Click
161.59.30.196→山西——→2021-06-01→1532057592622——→3572062651028442022→www.mi.com→View
98.173.26.105→香港——→2021-06-01→1532057592622——→9180539059907234209→www.jd.com→Click
79.48.85.87→广西——→2021-06-01→1532057592622——→527380487512630061→www.taobao.com→Login
123.180.29.190→辽宁——→2021-06-01→1532057592623——→3164374644101152078→www.suning.com→View
123.180.29.190→辽宁——→2021-06-01→1532057592623——→3164374644101152078→www.baidu.com→Buy
123.180.29.190→辽宁——→2021-06-01→1532057592623——→3164374644101152078→www.gome.com.cn→Regist
197.133.52.49→海南——→2021-06-01→1532057592624——→6064712595290452386→www.suning.com→Comment
```

图 8-1 电商网站用户访问日志数据集

由上图可知，数据集包含信息有：用户 IP 地址、用户 IP 所在省份/城市、用户访问日期、时间戳、序列号、访问网站地址、访问网站行为（如 Login、View 等）。

8.2.2 读取数据集

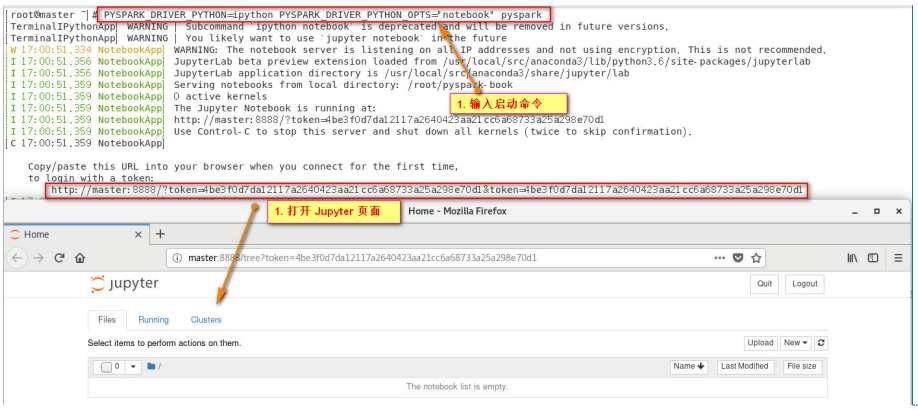
接下来，我们通过代码实战的方式进行读取并分析处理数据集。

步骤 1：启动 Pyspark Jupyter

启动虚拟机 Master 节点终端输入如下启动命令，启动 local 模式下的 Pyspark Notebook。

```
PYSPARK_DRIVER_PYTHON=ipython  PYSPARK_DRIVER_PYTHON_OPTS="notebook"  pyspark
```

在启动后，可以使用浏览器访问终端上显示的 jupyter 地址打开 jupyter 页面，如图 8-2 所示。



批注 [雷1]: 语意不明，需修改

批注 [雷2]: 图片内容需在正文中描写出来

批注 [HZ3R2]: 正文中有描述

图 8-2 jupyter notebook 页面

步骤 2：新建 Notebook

通过右上角【New】下拉菜单选择【Python3】，并确定，完成 Notebook 的新建工作，如图 8-3 所示。

图 8-3 新建 Notebook 文件

步骤 3：重命名 Notebook

首先单击【Untitled】，在弹出的【Rename Notebook】对话框中输入“Chapter08”，最后单击【Rename】按钮，为 Notebook 重命名，如图 8-4 所示。

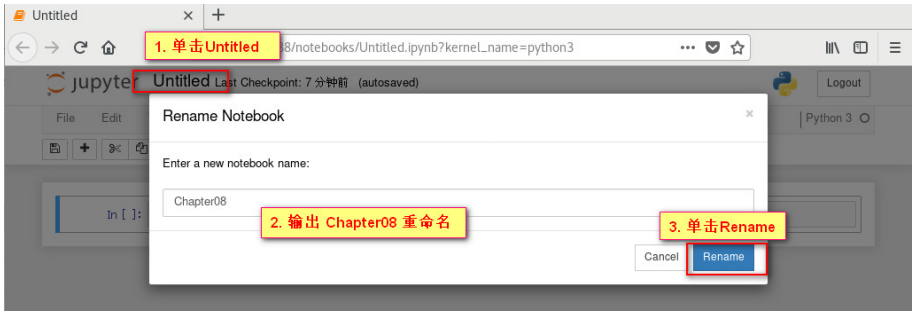


图 8-4 重命名 Notebook 文件

步骤 4：获取 Notebook 路径

在 Notebook 代码行中输入如下代码，获取 Notebook 所在路径并打印，如图 8-5 所示。

```
In [1]: pwd = !pwd
        data_path = "file://" + list(pwd)[0] + '/data/'
        print(data_path)

        file:///root/pyspark-book/data/
```

图 8-5 设置本地数据路径

> 温馨提示：
>
> Spark 读取本地文件需要在路径前加 file://，否则 Spark 默认读取的路径为 HDFS 上的地址，会报文件不存在的错误。若文件存在 HDFS 上，则直接使用文件路径。

步骤 5：导入数据集

我们使用 sc.textFile 读取 access_log.txt 数据文件，如图 8-6 所示。

```
In [2]: rdd = sc.textFile(data_path + "access_log.txt")
```

图 8-6 读取访问日志文件

读取的数据存储格式为 RDD。

步骤 6：查看数据集

我们通过 first()算子查看读取的数据集第一行数据，如图 8-7 所示。

```
In [3]: rdd.first()
Out[3]: '105.145.171.151\t辽宁\t2021-06-01\t1532057592615\t4389195290417379165\twww.jd.com\tView'
```

图 8-7 查看第一行数据

从以上结果可以看到，数据之间是通过“\t”作为分隔符连接各个字段。

8.2.3 PV 统计计算与分析

在上一小节中，我们通过代码读取了本地数据集，并进行了首行打印展示，了解了数据结构，本小节将通过代码实现对 PV 的统计计算。

步骤 1：编写 PV 计算函数

在 Notebook 代码行中创建 PV 计算函数，并编写 PV 计算逻辑代码，如图 8-8 所示。

```
In [5]: def pv(rows):
            sitepair = rows.map(lambda row: (row.split("\t")[5], 1))
            result = sitepair.reduceByKey(lambda v1, v2: (v1+v2))

            # 按照降序排序
            result = result.sortBy(lambda one: one[1], ascending=False)

            return result
```

图 8-8 PV 计算函数

通过该函数实现对 PV 的计算，代码命令实现详解如表 8-1 所示。

表 8-1 PV 计算函数主要命令

代码	代码含义
sitepair = rows.map(…)	使用 map 方法参数 rows 每一项数据进行转换
lambda row: …	lambda 匿名函数，传入每一项数据 row 作为参数进行处理
row.split("\t")[5], 1	将每一项数据以 Tab 键切分，将第 6 个字段（从 0 开始计数）组成(key, 1)组合
result = sitepair.reduceByKey(…)	将每个(key, 1)组合按 key 进行分组处理
lambda v1, v2: (v1+v2)	匿名函数，将 v1、v2 进行求和
result.sortBy(lambda one: one[1], ascending=False)	对 result 按每一项数据(匿名函数参数 one)的第一个字段(one[1])进行降序排序（ascending=False）

步骤 2：打印 PV 统计结果

执行 PV 统计函数并打印结果，PV 统计函数返回值类型仍为 RDD，所以可以直接使用 collect()算子进行结果展示，如图 8-9 所示。

```
In  [6]: pv(rdd).collect()

Out[6]: [('www.suning.com', 9),
         ('www.baidu.com', 9),
         ('www.gome.com.cn', 7),
         ('www.taobao.com', 5),
         ('www.dangdang.com', 5),
         ('www.jd.com', 4),
         ('www.mi.com', 2)]
```

图 8-9 查看 PV 统计结果

8.2.4 UV 统计计算与分析

在上一小节中，我们通过代码实现了 PV 的统计计算需求，在 8.1.1 小节中我们介绍了 PV 和 UV 的定义与区别，所以计算 UV 和计算 PV 的关键点在于去重的处理方法，本小节将通过代码实现对 UV 的统计计算。

步骤 1：编写 UV 计算函数

在 Notebook 代码行中创建 UV 计算函数，并编写 UV 计算逻辑代码，如图 8-10 所示。

```
In  [8]: def uv(rows):
             # 同一个ip，要distinct掉重
             sitepair = rows\
                 .map(lambda row: row.split("\t")[0] + "_" + row.split("\t")[5])\
                 .distinct()

             result = sitepair.map(lambda row: (row.split("_")[1], 1))\
                 .reduceByKey(lambda v1, v2: v1 + v2)\
                 .sortBy(lambda kv: kv[1], ascending=False)

             return result
```

图 8-10 UV 计算函数

通过该函数实现对 UV 的计算，UV 计算中关键点代码命令实现详解如表 8-2 所示。

表 8-2 UV 计算函数主要命令

代码	代码含义
sitepair = rows.map(…)	使用 map 方法参数 rows 每一项数据进行转换

lambda row: …	lambda 匿名函数，传入每一项数据 row 作为参数进行处理
row.split("\t")[0] + "_" + row.split("\t")[5]	将每一项数据以 Tab 键切割，并将第 1 个字段和第 6 个字段用下划线“_”连接
distinct()	进行去重
row.split("_")[1], 1)	将每一项数据以下划线“_”切分，并将第二个字段作为 key，组成(key, 1)组合

其余统计计算同 PV 计算逻辑，请参考表 8-1。

步骤 2：查看 UV 统计结果

执行 UV 统计函数并打印结果，UV 统计函数返回值类型仍为 RDD，所以可以直接使用 collect()算子进行结果展示，如图 8-11 所示。

```
In  [9]: uv(rdd).collect()
Out[9]: [('www.suning.com', 8),
         ('www.baidu.com', 8),
         ('www.gome.com.cn', 6),
         ('www.taobao.com', 5),
         ('www.jd.com', 4),
         ('www.dangdang.com', 3),
         ('www.mi.com', 2)]
```

图 8-11 查看 UV 统计结果

8.3 代码实现详解

在上一小节中，我们通过代码实现了关于 PV、UV 的计算，并对输出的 RDD 结果进行了打印展示。本小节将对上一小节中代码中关键算子和实现进行讲解。

本节代码中使用到的关键算子有 map、reduceByKey、distinct。关键算子的使用说明如表 8-3 所示。

表 8-3 算子使用介绍

算子	算子作用
map(func)	通过函数 func 传递源的每个元素，返回一个新的分布式数据集
reduceByKey(func)	在 (K, V) 对的数据集上调用时，返回 (K, V) 对的数据集，其中每个键的值使用给定的 reduce 函数 func 聚合
distinct	返回一个数据集元素不重复的新数据集

PV 计算实现如流程图 8-12 所示。

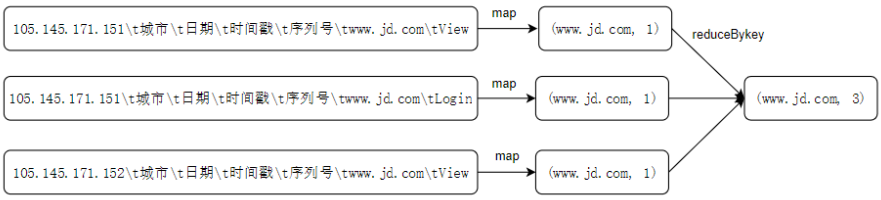


图 8-12 PV 计算代码流程图

UV 计算实现流程如图 8-13 所示。

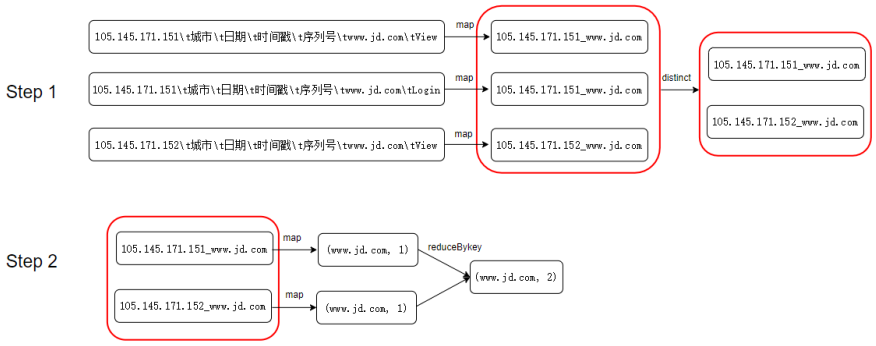


图 8-13 UV 计算代码流程图

由图 8-12 和图 8-13 观察可知，计算 UV 与计算 PV 相比，多进行了一步 map 转换和 distinct 去重，其中 map 转换的目的是为了转换成新的 key，即“IP_网站”形式，以该 key 进行数据去重的目的。

★回顾思考★

01　PySpark Notebook 启动 Local 模式的命令是什么？

答：

```
PYSPARK_DRIVER_PYTHON=ipython  PYSPARK_DRIVER_PYTHON_OPTS="notebook"  pyspark
```

批注 [雷4]: 图片步骤需在正文中介绍清楚

02　PV 和 UV 的含义是什么？

答：

（1） PV：PV（Page View）即页面浏览量或点击量，是网站分析的一个术语，是衡量一个网站或网页的用户访问量。

（2） UV：UV（Unique Visitor）即独立访客数，指访问某个站点或点击某个网页的不同 IP 地址的人数。

★练习★

一、选择题

1. Spark 读取本地文件，文件目录需要加的前缀是（ ）
   A. files://
   B. local://
   C. file://
   D. file:///
2. sc.TextFile 读取文件的格式为（ ）
   A. DatFrame
   B. RDD
   C. DataSet
   D. Text
3. RDD 使用哪些算子可以直接打印查看内容（ ）
   A. first()
   B. collect()
   C. print()
   D. show()

二、判断题

1. PV 是指某个站点不同 IP 的独立访客数。（×）
2. Spark 读取本地文件需要在路径前加前缀指定本地路径。（√）

三、实战练习

1. 将配套文件 access_log.txt 上传至 Linux 本地和 HDFS，分别读取计算 PV 值。

步骤 1：上传 access_log.txt 至 Linux 本地

可以通过 Xftp 等工具将文件上传至 Linux 本地或可以直接将存储在电脑本机上的文件通过拖拽文件的方式移动至虚拟机中。

步骤 2：上传 access_log.txt 至 HDFS

```
hdfs dfs -put Linux 本地路径/access_log.txt HDFS 路径/
```

步骤 3：计算 PV 值

请参考本章第 8.2.3 小节或本书配套 NoteBook 代码文件

提示：读取 HDFS 文件代码：

```
rdd = sc.textFile(HDFS 路径/ access_log.txt)
```

2. 将配套文件 access_log.txt 上传至 Linux 本地和 HDFS，分别读取计算 UV 值。

方法步骤同上，步骤 3 计算 UV 值请参考本章第 8.2.4 小节或本书配套 NoteBook 代码文件。

批注 [雷5]: 任务答案未完成，需解答

本章小结

本章详细介绍了电商领域常涉及的指标，并对 PV 和 UV 的定义进行了详细介绍。接着通过代码实战的方式实现了计算电商网站某日 PV、UV 值的项目，并对代码实现流程和代码实现关键算子的定义和使用进行了详细介绍。通过本章节的学习可以帮助读者加深对 RDD 算子的使用方法的了解，和进一

步理解大数据分析处理的广泛应用。