**大数据应用与服务赛项**

**任**

**务**

**书**

**1**

**背景描述**

大数据时代背景下，人们生活习惯发生了很多改变。在传统运营模式中，缺乏数据积累，人们在做出一些决策行为过程中，更多是凭借个人经验和直觉，发展路径比较自我封闭。而大数据时代，为人们提供一种全新的思路，通过大量的数据分析得出的结果将更加现实和准确。平台可以根据用户的浏览，点击，评论等行为信息数据进行收集和整理。通过大量用户的行为可以对某一个产品进行比较准确客观的评分和评价，或者进行相应的用户画像，将产品推荐给喜欢该产品的用户进行相应的消费。

因数据驱动的大数据时代已经到来，没有大数据，我们无法为用户提供大部分服务，为完成外卖网站的大数据分析工作，你所在的小组将应用大数据技术，通过Python语言将数据进行相应处理，并且进行数据标注、数据分析与可视化、通过大数据业务分析方法实现相应数据分析。运行维护数据库系统保障存储数据的安全性。通过运用大数据相关工具软件解决具体业务问题。你们作为该小组的技术人员，请按照下面任务完成本次工作。

## 模块一：平台搭建与运维

### 任务一：大数据平台搭建

#### 子任务一：Hadoop 完全分布式安装配置

本任务需要使用root用户完成相关配置，安装Hadoop需要配置前置环境。命令中要求使用绝对路径，具体要求如下:

1. 从Master中的/opt/software目录下将文件hadoop-3.1.3.tar.gz、jdk-8u191-linux-x64.tar.gz安装包解压到/opt/module路径中(若路径不存在，则需新建)，将JDK解压命令复制并粘贴至客户端桌面【Release\模块一提交结果.docx】中

mkdir -p /opt/module

cd /opt/software

mkdir -p /opt/module

tar -zxvf /opt/software/hadoop-3.1.3.tar.gz -C /opt/module/

tar -zxvf /opt/software/jdk-8u191-linux-x64.tar.gz -C /opt/module/

ls -la /opt/module/

1. 修改Master中/etc/profile文件，设置JDK环境变量并使其生效，配置完毕后在Master节点分别执行“java -version”和“javac”命令，将命令行执行结果分别截图并粘贴至客户端桌面【Release\模块一提交结果.docx】中对应的任务序号下；
2. 请完成host相关配置，将三个节点分别命名为master、slave1、slave2，并做免密登录，用scp命令并使用绝对路径从Master复制JDK解压后的安装文件到slave1、slave2节点（若路径不存在，则需新建），并配置slave1、slave2相关环境变量，将全部scp复制JDK的命令复制并粘贴至客户端桌面【Release\模块一提交结果.docx】中对应的任务序号下；
3. 在Master将Hadoop解压到/opt/module(若路径不存在，则需新建)目录下，并将解压包分发至slave1、slave2中，其中master、slave1、slave2节点均作为datanode，配置好相关环境，初始化Hadoop环境namenode，将初始化命令及初始化结果截图（截取初始化结果日志最后20行即可）粘贴至客户端桌面【Release\模块一提交结果.docx】中对应的任务序号下；
4. 启动Hadoop集群（包括hdfs和yarn），使用jps命令查看Master节点与slave1节点的Java进程，将jps命令与结果截图粘贴至客户端桌面【Release\模块一提交结果.docx】中对应的任务序号下。

#### 子任务二：Kafka安装配置

本任务需要使用root用户完成相关配置，已安装Hadoop及需要配置前置环境，具体要求如下：

1. 将Master节点ZooKeeper、Kafka安装包解压到/opt/module目录下，将Kafka解压命令复制并粘贴至客户端桌面【Release\模块一提交结果.docx】中对应的任务序号下；
2. 配置好ZooKeeper，其中ZooKeeper使用集群模式，分别将master、slave1、slave2作为其节点（若ZooKeeper已安装配置好，则无需再次配置），配置好Kafka的环境变量，使用kafka-server-start.sh --version查看Kafka的版本内容，并将命令和结果截图粘贴至客户端桌面【Release\模块一提交结果.docx】中对应的任务序号下；
3. 完善其他配置并分发Kafka文件到slave1、slave2中，并在每个节点启动Kafka，创建Topic，其中Topic名称为installtopic，分区数为2，副本数为2，将创建命令和创建成果截图粘贴至客户端桌面【Release\模块一提交结果.docx】中对应的任务序号下。

#### 子任务三：Flink on Yarn安装配置

本任务需要使用root用户完成相关配置，已安装Hadoop及需要配置前置环境，具体要求如下：

1. 从Master中的/opt/software目录下将文件flink-1.14.0-bin-scala\_2.12.tgz解压到路径/opt/module中(若路径不存在，则需新建)，将完整解压命令复制粘贴至客户端桌面【Release\模块一提交结果.docx】中对应的任务序号下；
2. 修改Master中/etc/profile文件，设置Flink环境变量并使环境变量生效。在容器中/opt目录下运行命令flink --version，将命令与结果截图粘贴至客户端桌面【Release\模块一提交结果.docx】中对应的任务序号下；
3. 开启Hadoop集群，在yarn上以per job模式（即Job分离模式，不采用Session模式）运行 $FLINK\_HOME/examples/batch/WordCount.jar，将运行结果最后10行截图粘贴至客户端桌面【Release\模块一提交结果.docx】中对应的任务序号下。

示例 ：

flink run -m yarn-cluster -p 2 -yjm 2G -ytm 2G $FLINK\_HOME/examples/batch/WordCount.jar

### 任务二：数据库配置维护

#### 子任务一：数据库配置

1、配置服务端MySQL数据库的远程连接。

2、初始化MySQL数据库系统，将完整命令及初始化成功的截图复制粘贴至客户端桌面【Release\模块一提交结果.docx】中对应的任务序号下。

3、配置root用户允许任意ip连接，将完整命令截图复制粘贴至客户端桌面【Release\模块一提交结果.docx】中对应的任务序号下。

4、通过root用户登录MySQL数据库系统，查看mysql库下的所有表，将完整命令及执行命令后的结果的截图复制粘贴至客户端桌面【Release\模块一提交结果.docx】中对应的任务序号下。

5、输入命令以创建新的用户。完整命令及执行命令后的结果的截图复制粘贴至客户端桌面【Release\模块一提交结果.docx】中对应的任务序号下。

6、授予新用户访问数据的权限。完整命令及执行命令后的结果的截图复制粘贴至客户端桌面【Release\模块一提交结果.docx】中对应的任务序号下。

7、刷新权限。完整命令及执行命令后的结果的截图复制粘贴至客户端桌面【Release\模块一提交结果.docx】中对应的任务序号下。

#### 子任务二：创建相关表

1. 根据以下数据字段在MySQL数据库中创建店铺表（shop）。店铺表字段如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 中文含义 | 备注 |
| id | int | 店铺编号 |  |
| shop\_name | varchar | 店铺名称 |  |
| city | varchar | 城市 |  |
| location | varchar | 商圈 |  |
| longtitude | double | 经度 |  |
| latitude | double | 纬度 |  |
| shop\_type | varchar | 商铺类型 |  |
| pay\_cnt | bigint | 订单数 |  |

1. 根据以下数据字段在MySQL数据库中创建评论表（comment）。评论表字段如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 中文含义 | 备注 |
| id | int | 评论编号 |  |
| shop\_name | varchar | 店铺名称 |  |
| commentator | varchar | 评论人 |  |
| score | double | 评分 |  |
| comment\_time | datetime | 评论时间 |  |
| content | varchar | 评论内容 |  |

将这两个SQL建表语句分别截图复制粘贴至客户端桌面【Release\模块一提交结果.docx】中对应的任务序号下。

#### 子任务三：维护数据表

根据已给到的sql文件将这两份数据导入任意自己创建的数据库中，并对其中的数据进行如下操作：

1. 在shop\_all表中删除id为25的店铺数据;
2. 在comment\_all表中将id为30的评分改为5;
3. 在shop\_all表中统计各商圈的店铺总数。

将这三个SQL语句分别截图复制粘贴至客户端桌面【Release\模块一提交结果.docx】中对应的任务序号下。

## 模块二：数据获取与处理

### 任务一：数据获取与清洗

现已从相关网站及平台获取到原始数据集，为保障用户隐私和行业敏感信息，已进行数据脱敏。数据脱敏是指对某些敏感信息通过脱敏规则进行数据的变形，实现敏感隐私数据的可靠保护。在涉及客户安全数据或者一些商业性敏感数据的情况、不违反系统规则条件下，对真实数据进行改造并提供测试使用，如身份证号、手机号等个人信息都需要进行数据脱敏。

相关数据文件中已经包含了数据采集阶段从餐饮外送统计平台网站上爬取的数据集，，你的小组需要通过编写代码或脚本完成对相关数据文件中数据的清洗和整理。

请分析相关数据集，根据题目规定要求实现数据处理，具体要求如下：

1. 删除distribution.csv中**配送取消异常数（exception\_delivery\_cancellations\_num）**为空的数据并且存入distribution2\_c1\_N.csv ，N为删除的数据条数；
2. 删除distribution.csv中缺失值大于3个的数据列并且存入distribution2\_c2\_N.csv， N为删除的数据列数；
3. 将distribution.csv中**系统取消数（system\_cancellations\_num）**为空的数据设置为0并且存入distribution2\_c3.csv；
4. 将distribution.csv中**整体时长（overall\_duration）**为空的数据设置为平均时长（保留一位小数）并且存入distribution2\_c4\_N.csv,N为平均评分。

将该4个文件名截一张图复制粘贴至客户端桌面【Release\模块二提交结果.docx】中对应的任务序号下。

### 任务二：数据标注

使用SnowNLP对店铺评论数据comment.csv进行标注，获取情感倾向评分（sentiments），具体的标注规则如下：

1. 对情感倾向分数大于等于0.6的评论数据标注为正向；
2. 对情感倾向分数大于0.4小于0.6的数据标注为中性；
3. 对情感倾向分数小于等于0.4的评论信息标注为负向。

根据采集到的评论信息，给出三类标注好的数据，存入standard.csv。将standard.csv打开后直接截图（至少显示前12条）复制粘贴至客户端桌面【Release\模块二提交结果.docx】中对应的任务序号下。

具体格式如下：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 编号 | 店铺名称 | 评论 | 用户名 | 评价日期 | 评分 | 情感倾向 |
| 0 | RoyBlteB皇茶（广厦店） | 很快 | 辉\*\*婷 | 2022/2/15 | 4 | 正向 |

### 任务三：数据统计

#### 子任务一：HDFS文件上传下载

本任务需要使用Hadoop、HDFS命令，已安装Hadoop及需要配置前置环境，具体要求如下：

1. 在Master中的/root/目录下新建一个文件夹:result，将创建文件夹命令与结果截图粘贴至客户端桌面【Release\模块二提交结果.docx】中对应的任务序号下；
2. 使用HDFS命令，将Master下：/root目录下新建的文件夹：result上传到HDFS指定目录下：/根目录下；并且使用HDFS命令查看目录；将HDFS上传，查看命令截图粘贴至客户端桌面【Release\模块二提交结果.docx】中对应的任务序号下；
3. 使用HDFS命令，将HDFS目录下的/result文件夹下载到Master指定目录下:根目录下；将下载文件夹命令与结果截图粘贴至客户端桌面【Release\模块二提交结果.docx】中对应的任务序号下。

#### 子任务二：数据统计

user\_info.csv 文件存储了电商互联网平台上收集的用户数据，数据中有以下内容：

id：主键非空，bigint类型，长度为 20

login\_name：用户名，varchar类型，长度200

nick\_name：用户昵称，varchar类型，长度200

passwd：密码，varchar类型，长度200

name：姓名，varchar类型，长度200

phone\_num:手机号，varchar类型，长度200

email:邮箱，varchar类型，长度200

head\_img:头像，varchar类型，长度200

user\_level:用户级别，varchar类型，长度200

birthday:用户生日，date类型，长度0，格式为YYYY-MM-DD

gender:性别，varchar类型，长度1

create\_time:创建时间，datetime类型，格式为 yyyy-MM-dd HH:mm:ss

operate\_time:修改时间，datetime类型，格式为 yyyy-MM-dd HH:mm:ss

编写 MapReduce 程序，实现以下功能：将user\_info.csv数据的分隔符“，”转换为“|”，输出文件到HDFS，然后在在控制台按顺序打印输出前 10 条数据，将结果截图粘贴至客户端桌面【Release\模块二提交结果.docx】中对应的任务序号下。

## 模块三：业务分析与可视化

### 任务一：数据可视化

使用Python代码编写数据可视化的相关功能，所用数据为poi\_all.csv数据，具体要求如下：

1. 用条形图显示各个城市的店铺总数（不必刻意排序）；
2. 用条形图显示各个城市**30天销量（A平台）（A\_day\_30\_cnt）**的平均值（不必刻意排序）；
3. 用饼图显示各平台店铺数占比。

将该3个可视化图表分别截图复制粘贴至客户端桌面【Release\模块三提交结果.docx】中对应的任务序号下。

### 任务二：业务分析

1、根据模块二的结果数据standard.csv评论情感分析结果，以月度为单位统计每月所有店铺的正向、中性、负向评价数量，绘制折线图，并对店铺的发展趋势作出简要分析。将图表截图复制粘贴至客户端桌面【Release\模块三提交结果.docx】中对应的任务序号下，并在其下方编写发展趋势分析。

2、根据standard.csv文件中的结果，通过excel 生成报表信息方便店铺在后续服务中进行优化，及时准确的把握市场行情，具体要求如下：

1. 所有店铺的评论正向、负向和中性的评论趋势；
2. 所有店铺在互联网上的整体评价趋势饼图。

将两张图表截一张图复制粘贴至客户端桌面【Release\模块三提交结果.docx】中对应的任务序号下。

### 附：名词变量名对应关系表

|  |  |
| --- | --- |
| **名词** | **变量名** |
| （平台）商铺id | Walle\_id |
| （配送系统）商铺id | Retailer\_id |
| 商铺名称 | retailer\_name |
| 商铺地址 | retailer\_address |
| 位置POI编码 | retailer\_location |
| 城市id | City\_id |
| 网格id | Grid\_id |
| 渠道经理 | Carrier\_id |
| 渠道小组 | Team\_id |
| 申请人id | Applicant\_id |
| 申请人 | Applicant\_name |
| 一级审批角色 | first\_auditor\_role |
| 候选审批人id | first\_auditor\_candidate\_ids |
| 审核人id | first\_auditor\_id |
| 审批人名 | first\_auditor\_name |
| 二级审批角色 | second\_auditor\_role |
| 审核小组成员 | second\_auditor\_candidate\_ids |
| 二级审核人id | second\_auditor\_id |
| 二级审核人名 | second\_auditor\_name |
| 申请状态 | status |
| 申请前最大配送距离 | max\_distance\_before\_edit |
| 申请前最小配送距离 | min\_distance\_before\_edit |
| 申请后最大配送距离 | max\_distance\_after\_edit |
| 申请后最小配送距离 | min\_distance\_after\_edit |
| 申请前配送面积 | area\_before\_edit |
| 申请前配送面积 | area\_after\_edit |
| 申请递交时间 | application\_submission\_time |
| 审核完成 | application\_completion\_time |
| 申请时间 | application\_time |
| 推单数(9日) | push\_single9 |
| 有效完成率(9日) | effective\_completion\_rate9 |
| 超时率(9日) | timeout\_rate9 |
| 推单数(8日) | push\_single8 |
| 有效完成率(8日) | effectiv\_completion\_rate8 |
| 超时率(8日) | timeout\_rate8 |
| 城市 | city\_name |
| 商圈 | location |
| 纬度 | latitude |
| 经度 | longtitude |
| 商家所属平台 | rest\_type |
| 商家id（A平台） | platform\_A\_restid |
| 店铺名称（A平台） | A\_rst\_name |
| 30天销量（A平台） | A\_day\_30\_cnt |
| 商家id（B平台） | platform\_B\_restid |
| 店铺名称（B平台） | B\_rst\_name |
| 30天销量（B平台） | B\_day\_30\_cnt |
| 接单数 | orders\_num |
| 有效完成单数 | complete\_orders\_num |
| 投诉率 | complaint\_rate |
| 异常率 | abnormal\_rate |
| 欺诈单数 | faker\_orders\_num |
| 拒单数 | refuse\_orders\_num |
| 商户取消数 | merchant\_cancellation\_num |
| 客户取消数 | customer\_cancellations\_num |
| 系统取消数 | system\_cancellations\_num |
| 配送取消异常数 | exception\_delivery\_cancellations\_num |
| 整体时长 | overall\_duration |
| 接单时长 | order\_time |
| 到店时长 | time\_to\_shop |
| 取餐时长 | take\_meal\_time |
| 送达时长 | delivery\_time |
| 商户投诉数 | merchant\_complaints\_num |
| 用户投诉数 | user\_complaints\_num |
| 差评数 | negative\_reviews\_num |
| 好评数 | positive\_reviews\_num |
| 评价数 | evaluation\_num |
| 最远订单距离 | farthest\_order\_distance |
| 商户业务包 | merchant\_business\_package |

### 附：matplotlib和seaborn解决中文显示问题

import seaborn

import matplotlib.pyplot as plt

plt.rcParams['font.sans-serif'] = ['SimHei'] # 黑体

plt.rcParams['axes.unicode\_minus'] = False # 解决无法显示符号的问题

seaborn.set(font='SimHei', font\_scale=0.8) # 解决Seaborn中文显示问题