2022年全国职业院校技能大赛

高职组

“大数据技术与应用”

赛项赛卷（6卷）

任

务

书

参赛队编号：

**背景描述**

工业互联网是工业全要素、全产业链、全价值链的全面连接，是人、机、物、工厂互联互通的新型工业生产制造服务体系，是互联网从消费领域向生产领域、从虚拟经济向实体经济拓展的核心载体，是建设现代化经济体系、实现高质量发展和塑造全球产业竞争力的关键支撑。党中央、国务院高度重视工业互联网发展，习近平总书记连续四年对推动工业互联网发展做出重要指示。加快发展工业互联网产业，不仅是各国顺应产业发展大势，抢占产业未来制高点的战略选择，也是我国推动制造业质量变革、效率变革和动力变革，实现高质量发展的客观要求。

为完成工业大数据分析工作，你所在的小组将应用大数据技术，以Scala作为整个项目的基础开发语言，基于大数据平台综合利用

Hive、Spark、Flink、Vue.js等技术，对数据进行处理、分析及可视化呈现，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A：大数据平台搭建（容器环境）（15分）

**环境说明：**

|  |
| --- |
| **服务端登录地址详见各模块服务端说明。**  **补充说明：**宿主机可通过Asbru工具或SSH客户端进行SSH访问；  相关软件安装包在宿主机的/opt目录下，请选择对应的安装包进行安装，用不到的可忽略；  所有模块中应用命令必须采用绝对路径；  进入Master节点的方式为  docker exec –it master /bin/bash  进入Slave1节点的方式为  docker exec –it slave1 /bin/bash  进入Slave2节点的方式为  docker exec –it slave2 /bin/bash  MySQL已在容器的Master中安装完毕，用户名/密码为root/123456 |

### 任务一：Hadoop 完全分布式安装配置

本环节需要使用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. 从宿主机/opt目录下将文件hadoop-2.7.7.tar.gz、jdk-8u212-linux-x64.tar.gz复制到容器master中的/opt/software路径中（若路径不存在，则需新建），将master节点JDK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对应报告中；
2. 修改容器中/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对应报告中；
3. 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对应报告中；
4. 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）粘贴至对应报告中；
5. 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对应报告中。

### 任务二：Sqoop安装配置

本环节需要使用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. 从宿主机/opt目录下将文件apache-hive-2.3.4-bin.tar.gz、mysql-connector-java-5.1.47.jar复制到容器master中的/opt/software路径中（若路径不存在，则需新建）, 将容器master节点Hive安装包解压到/opt/module目录下，将命令复制并粘贴至对应报告中；
2. 完善其他Sqoop相关配置，设置Sqoop环境变量，并使环境变量生效，执行命令sqoop version并将命令与其结果截图粘贴至对应报告中；
3. 测试Sqoop连接容器master节点的MySQL数据库是否成功并展示所有的database，并将连接命令与结果截图粘贴至对应报告中。

### 任务三：Hive安装配置

本环节需要使用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. 从宿主机/opt目录下将文件apache-hive-2.3.4-bin.tar.gz、mysql-connector-java-5.1.47.jar复制到容器master中的/opt/software路径中（若路径不存在，则需新建），将Master节点Hive安装包解压到/opt/module目录下，将命令复制并粘贴至对应报告中；
2. 设置Hive环境变量，并使环境变量生效，执行命令hive --version并将命令与结果截图容复制并粘贴至对应报告中；
3. 完成相关配置并添加所依赖包，将MySQL数据库作为Hive元数据库。初始化Hive元数据，并通过schematool相关命令执行初始化，将初始化结果截图（范围为命令执行结束的最后10行）复制粘贴至对应报告中。

## 模块B：离线数据处理（25分）

**环境说明：**

|  |
| --- |
| **服务端登录地址详见各模块服务端说明。**  **补充说明：**各主机可通过Asbru工具或SSH客户端进行SSH访问；  Master节点MySQL数据库用户名/密码：root/123456（已配置远程连接）；  Hive的配置文件位于/opt/apache-hive-2.3.4-bin/conf/  Spark任务在Yarn上用Client运行，方便观察日志。  **注：该Spark版本无法进行本地调试，请打包上传集群调试。** |

### 任务一：数据抽取

编写Sqoop脚本，将MySQL库中表ChangeRecord，BaseMachine，MachineData， ProduceRecord全量抽取到Hive的ods库中对应表changerecord，basemachine， machinedata，producerecord中。（提示：可通过sqoop将mysql的数据先加载到hdfs，然后再通过hive中load data inpath的方式为将数据加载到分区表中，同时hive表中默认的分隔符为\t）

1. 抽取MySQL的shtd\_industry库中ChangeRecord表的全量数据进入Hive的ods库中表changerecord，字段排序、类型不变，同时添加静态分区，分区字段类型为String，且值为当前比赛日的前一天日期（分区字段格式为yyyyMMdd）。使用hive cli执行show partitions ods.changerecord命令，将Sqoop提交命令及hive cli的执行结果分别截图粘贴至对应报告中；
2. 抽取MySQL的shtd\_industry库中BaseMachine表的全量数据进入Hive的ods库中表basemachine，字段排序、类型不变，同时添加静态分区，分区字段类型为String，且值为当前比赛日的前一天日期（分区字段格式为yyyyMMdd）。使用hive cli执行show partitions ods.basemachine命令，将Sqoop提交命令及hive cli的执行结果分别截图粘贴至对应报告中；
3. 抽取MySQL的shtd\_industry库中ProduceRecord表的全量数据进入Hive的ods库中表producerecord，剔除ProducePrgCode字段，其余字段排序、类型不变，同时添加静态分区，分区字段类型为String，且值为当前比赛日的前一天日期（分区字段格式为yyyyMMdd）。使用hive cli执行show partitions ods.producerecord命令，将Sqoop提交命令及hive cli的执行结果分别截图粘贴至对应报告中；
4. 抽取MySQL的shtd\_industry库中MachineData表的全量数据进入Hive的ods库中表machinedata，字段排序、类型不变，同时添加静态分区，分区字段类型为String，且值为当前比赛日的前一天日期（分区字段格式为yyyyMMdd）。使用hive cli执行show partitions ods.machinedata命令，将Sqoop提交命令及hive cli的执行结果分别截图粘贴至对应报告中。

### 任务二：数据清洗

编写Hive SQL代码，将ods库中相应表数据全量抽取到Hive的dwd库中对应表中。表中有涉及到timestamp类型的，均要求按照yyyy-MM-dd HH:mm:ss，不记录毫秒数，若原数据中只有年月日，则在时分秒的位置添加00:00:00，添加之后使其符合yyyy-MM-dd HH:mm:ss。

1. 抽取ods库中changerecord的全量数据进入Hive的dwd库中表fact\_change\_record，抽取数据之前需要对数据根据changeid和changemachineid进行联合去重处理，分区字段为etldate且值与ods库的相对应表该值相等，并添加dwd\_insert\_user、dwd\_insert\_time、dwd\_modify\_user、dwd\_modify\_time四列,其中dwd\_insert\_user、dwd\_modify\_user均填写“user1”，dwd\_insert\_time、dwd\_modify\_time均填写当前操作时间，并进行数据类型转换。使用hive cli按照change\_machine\_id降序排序，查询前1条数据，将结果截图粘贴至对应报告中；
2. 抽取ods库中basemachine的全量数据进入Hive的dwd库中表dim\_machine,抽取数据之前需要对数据根据basemachineid进行去重处理。分区字段为etldate且值与ods库的相对应表该值相等，并添加dwd\_insert\_user、dwd\_insert\_time、dwd\_modify\_user、dwd\_modify\_time四列,其中dwd\_insert\_user、dwd\_modify\_user均填写“user1”，dwd\_insert\_time、dwd\_modify\_time均填写当前操作时间，并进行数据类型转换。使用hive cli按照base\_machine\_id顺序排序，查询dim\_machine前2条数据，将结果截图粘贴至对应报告中；
3. 抽取ods库中producerecord的全量数据进入Hive的dwd库中表fact\_produce\_record,分区字段为etldate且值与ods库的相对应表该值相等，并添加dwd\_insert\_user、dwd\_insert\_time、dwd\_modify\_user、dwd\_modify\_time四列,其中dwd\_insert\_user、dwd\_modify\_user均填写“user1”，dwd\_insert\_time、dwd\_modify\_time均填写当前操作时间，并进行数据类型转换。使用hive cli按照produce\_machine\_id升序排序，查询fact\_produce\_record前1条数据，将结果截图粘贴至对应报告中；
4. 抽取ods库中machinedata的全量数据进入Hive的dwd库中表fact\_machine\_data。分区字段为etldate且值与ods库的相对应表该值相等，并添加dwd\_insert\_user、dwd\_insert\_time、dwd\_modify\_user、dwd\_modify\_time四列,其中dwd\_insert\_user、dwd\_modify\_user均填写“user1”，dwd\_insert\_time、dwd\_modify\_time均填写当前操作时间，并进行数据类型转换。使用hive cli按照machine\_id降序排序，查询前1条数据，将结果截图粘贴至对应报告中。

### 任务三：指标计算

1. 编写Scala代码，使用Spark根据dwd层的fact\_change\_record表统计每个月（change\_start\_time的月份）、每个设备、每种状态的时长，若某状态当前未结束（即change\_end\_time值为空）则该状态不参与计算。计算结果存入MySQL数据库shtd\_industry的machine\_state\_time表中（表结构如下），然后在Linux的MySQL命令行中根据设备id、状态持续时长均为降序排序，查询出前10条，将SQL语句与执行结果截图粘贴至对应报告中;

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 中文含义 | 备注 |
| machine\_id | int | 设备id |  |
| change\_record\_state | varchar | 状态 |  |
| duration\_time | varchar | 持续时长（秒） | 当月该状态的时长和 |
| year | int | 年 | 状态产生的年 |
| month | int | 月 | 状态产生的月 |

1. 编写Scala代码，使用Spark根据dwd层的fact\_change\_record表关联dim\_machine表统计每个车间中所有设备运行时长（即设备状态为“运行”）的中位数在哪个设备（为偶数时，两条数据原样保留输出），若某个设备运行状态当前未结束（即change\_end\_time值为空）则该状态不参与计算，计算结果存入MySQL数据库shtd\_industry的machine\_running\_median表中（表结构如下），然后在Linux的MySQL命令行中根据所属车间、设备id均为降序排序，查询出前5条数据，将SQL语句与执行结果截图粘贴至对应报告中;

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 中文含义 | 备注 |
| machine\_id | int | 设备id |  |
| machine\_factory | int | 所属车间 |  |
| total\_running\_time | int | 运行总时长 | 结果以秒为单位 |

1. 编写Scala代码，使用Spark根据dwd层的fact\_produce\_record表，基于全量历史数据计算各设备生产一个产品的平均耗时，produce\_code\_end\_time值为1900-01-01 00:00:00的数据为脏数据，需要剔除，并以produce\_record\_id和produce\_machine\_id为联合主键进行去重（注：fact\_produce\_record表中，一条数据代表加工一个产品，produce\_code\_start\_time字段为开始加工时间，produce\_code\_end\_time字段为完成加工时间），将设备每个产品的耗时与该设备平均耗时作比较，保留耗时高于平均值的产品数据，将得到的数据写入MySQL数据库shtd\_industry的machine\_produce\_per\_avgtime表中（表结构如下），然后在Linux的MySQL命令行中根据设备id降序排序查询前3条数据，将SQL语句与执行结果截图粘贴至对应报告中。

machine\_produce\_per\_avgtime：

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 中文含义 | 备注 |
| produce\_record\_id | int | 生产记录id | 每生产一件产品产生一条数据 |
| produce\_machine\_id | int | 设备id |  |
| producetime | int | 该产品耗时 |  |
| produce\_per\_avgtime | int | 设备生产一个产品平均耗时 | 单位：秒 |

## 模块C：数据挖掘（10分）

**环境说明：**

|  |
| --- |
| **服务端登录地址详见各模块服务端说明。**  **补充说明：**各主机可通过Asbru工具或SSH客户端进行SSH访问；  Master节点MySQL数据库用户名/密码：root/123456（已配置远程连接）；  Hive的配置文件位于/opt/apache-hive-2.3.4-bin/conf/  Spark任务在Yarn上用Client运行，方便观察日志。  该模块均使用Scala编写，利用Spark相关库完成。  **注：该Spark版本无法进行本地调试，请打包上传集群调试。** |

### 任务一：特征工程

1. 根据dwd库中fact\_machine\_data表（或MySQL的shtd\_industry库中MachineData表），根据以下要求转换：获取最大分区（MySQL不用考虑）的数据后，首先解析列machine\_record\_data（MySQL中为MachineRecordData）的数据（数据格式为xml，采用dom4j解析，会给出解析demo），并获取每条数据的主轴转速，主轴倍率，主轴负载，进给倍率，进给速度，PMC程序号，循环时间，运行时间，有效轴数，总加工个数，已使用内存，未使用内存，可用程序量，注册程序量等相关的值（若该条数据没有相关值，则按下表设置默认值），同时转换machine\_record\_state字段的值，若值为报警，则填写1，否则填写0，以下为表结构，将数据保存在dwd.fact\_machine\_learning\_data，使用hive cli按照machine\_record\_id升序排序，查询dwd.fact\_machine\_learning\_data前1条数据，将结果截图粘贴至对应报告中。

dwd.fact\_machine\_learning\_data表结构：

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 中文含义 | 备注 |
| machine\_record\_id | int | 自增长id |  |
| machine\_id | double | 机器id |  |
| machine\_record\_state | double | 机器状态 | 默认0.0 |
| machine\_record\_mainshaft\_speed | double | 主轴转速 | 默认0.0 |
| machine\_record\_mainshaft\_multiplerate | double | 主轴倍率 | 默认0.0 |
| machine\_record\_mainshaft\_load | double | 主轴负载 | 默认0.0 |
| machine\_record\_feed\_speed | double | 进给倍率 | 默认0.0 |
| machine\_record\_feed\_multiplerate | double | 进给速度 | 默认0.0 |
| machine\_record\_pmc\_code | double | PMC程序号 | 默认0.0 |
| machine\_record\_circle\_time | double | 循环时间 | 默认0.0 |
| machine\_record\_run\_time | double | 运行时间 | 默认0.0 |
| machine\_record\_effective\_shaft | double | 有效轴数 | 默认0.0 |
| machine\_record\_amount\_process | double | 总加工个数 | 默认0.0 |
| machine\_record\_use\_memory | double | 已使用内存 | 默认0.0 |
| machine\_record\_free\_memory | double | 未使用内存 | 默认0.0 |
| machine\_record\_amount\_use\_code | double | 可用程序量 | 默认0.0 |
| machine\_record\_amount\_free\_code | double | 注册程序量 | 默认0.0 |
| machine\_record\_date | timestamp | 记录日期 |  |
| dwd\_insert\_user | string |  |  |
| dwd\_insert\_time | timestamp |  |  |
| dwd\_modify\_user | string |  |  |
| dwd\_modify\_time | timestamp |  |  |

### 任务二：报警预测

1. 根据任务一的结果，建立随机森林（随机森林相关参数可自定义，不做限制），使用任务一的结果训练随机森林模型，然后再将hive中dwd.fact\_machine\_learning\_data\_test（该表字段含义与dwd.fact\_machine\_learning\_data表相同，machine\_record\_state列值为空，表结构自行查看）转成向量，预测其是否报警将结果输出到MySQL数据库shtd\_industry中的ml\_result表中（表结构如下）。在Linux的MySQL命令行中查询出machine\_record\_id为1、8、20、28和36的5条数据，将SQL语句与执行结果截图粘贴至对应报告中。

ml\_result表结构：

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 中文含义 | 备注 |
| machine\_record\_id | int | 自增长id |  |
| machine\_record\_state | double | 机器状态 | 报警为1，其他状态则为0 |

## 模块D：数据采集与实时计算（20分）

**环境说明：**

|  |
| --- |
| **服务端登录地址详见各模块服务端说明。**  **补充说明：**各主机可通过Asbru工具或SSH客户端进行SSH访问；  请先检查ZooKeeper、Kafka、Redis端口是否已启动，若未启动则各启动命令如下：  ZK启动（netstat -ntlp查看2181端口是否打开）  /usr/zk/zookeeper-3.4.6/bin/zkServer.sh start  Redis启动（netstat -ntlp查看6379端口是否打开）  /usr/redis/bin/redis-server /usr/redis/bin/redis.conf  Kafka启动（netstat -ntlp查看9092端口是否打开）  /opt/kafka/kafka\_2.11-2.0.0/bin/kafka-server-start.sh -daemon（空格连接下一行）/opt/kafka/kafka\_2.11-2.0.0/config/server.properties  Flink任务在Yarn上用per job模式（即Job分离模式，不采用Session模式），方便Yarn回收资源。 |

### 任务一：实时数据采集

1. 在Master节点使用Flume采集/data\_log目录下实时日志文件中的数据，将数据存入到Kafka的Topic中（Topic名称分别为ChangeRecord、ProduceRecord和EnvironmentData，分区数为4），将Flume采集ChangeRecord主题的配置截图粘贴至对应报告中；
2. 编写新的Flume配置文件，将数据备份到HDFS目录/user/test/flumebackup下，要求所有主题的数据使用同一个Flume配置文件完成，将Flume的配置截图粘贴至对应报告中。

### 任务二：使用Flink处理Kafka中的数据

编写Scala工程代码，使用Flink消费Kafka中的数据并进行相应的数据统计计算。

1. 使用Flink消费Kafka中ProduceRecord主题的数据，统计在已经检验的产品中，各设备每5分钟生产产品总数，将结果存入Redis中，key值为“totalproduce”，value值为“设备id，最近5分钟生产总数”。使用redis cli以HGETALL key方式获取totalproduce值，将结果截图粘贴至对应报告中，需两次截图，第一次截图和第二次截图间隔5分钟以上，第一次截图放前面，第二次截图放后面；

注：ProduceRecord主题，生产一个产品产生一条数据；

change\_handle\_state字段为1代表已经检验，0代表未检验；

时间语义使用Processing Time。

1. 使用Flink消费Kafka中ChangeRecord主题的数据，当某设备30秒状态连续为“预警”，输出预警信息。当前预警信息输出后，最近30秒不再重复预警（即如果连续1分钟状态都为“预警”只输出两次预警信息），将结果存入Redis中，key值为“warning30sMachine”，value值为“设备id，预警信息”。使用redis cli以HGETALL key方式获取warning30sMachine值，将结果截图粘贴至对应报告中，需两次截图，第一次截图和第二次截图间隔1分钟以上，第一次截图放前面，第二次截图放后面；

注：时间使用change\_start\_time字段，忽略数据中的change\_end\_time不参与任何计算。忽略数据迟到问题。

Redis的value示例：115,2022-01-01 09:53:10:设备115 连续30秒为预警状态请尽快处理！

(2022-01-01 09:53:10 为change\_start\_time字段值，中文内容及格式必须为示例所示内容。)

1. 使用Flink消费Kafka中ChangeRecord主题的数据，统计每3分钟各设备状态为“预警”且未处理的数据总数，将结果存入MySQL数据库shtd\_industry的threemin\_warning\_state\_agg表中（追加写入，表结构如下）。请将任务启动命令截图，启动且数据进入后按照设备id降序排序查询threemin\_warning\_state\_agg表进行截图，第一次截图后等待3分钟再次查询并截图,将结果截图粘贴至对应报告中。

threemin\_warning\_state\_agg表：

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 中文含义 |
| change\_machine\_id | int | 设备id |
| totalwarning | int | 未被处理预警的数据总数 |
| window\_end\_time | varchar | 窗口结束时间（yyyy-MM-dd HH:mm:ss） |

注：时间语义使用Processing Time。

## 模块E：数据可视化（15分）

**环境说明：**

|  |
| --- |
| **数据接口地址及接口描述详见各模块服务端说明。** |

### 任务一：用柱状图展示设备历史各个状态持续时长

编写Vue工程代码，根据接口，用柱状图展示接口所有数据中各设备各个状态持续时长（秒），同时将用于图表展示的数据结构在浏览器的console中进行打印输出，将图表可视化结果和浏览器console打印结果分别截图并粘贴至对应报告中。

### 任务二：用柱状图展示每日所有车间各设备平均运行时长

编写Vue工程代码，根据接口，用柱状图展示每日所有车间各设备平均运行时长（秒），同时将用于图表展示的数据结构在浏览器的console中进行打印输出，将图表可视化结果和浏览器console打印结果分别截图并粘贴至对应报告中。

任务三：用折线图展示机器运行时长

编写Vue工程代码，根据接口，用折线图展示机器OP160每日的运行时长（秒），同时将用于图表展示的数据结构在浏览器的console中进行打印输出，将图表可视化结果和浏览器console打印结果分别截图并粘贴至对应报告中。

任务四：用条形图展示每日各设备产量

编写Vue工程代码，根据接口，用条形图展示每日各设备产量，同时将用于图表展示的数据结构在浏览器的console中进行打印输出，将图表可视化结果和浏览器console打印结果分别截图并粘贴至对应报告中。

任务五：用折柱混合图展示设备日均产量和所在车间日均产量

编写Vue工程代码，根据接口，用折柱混合图展示设备日均产量和所在车间日均产量，其中柱状图展示各设备的日均产量，折线图展示该设备所在车间的日均产量，同时将用于图表展示的数据结构在浏览器的console中进行打印输出，将图表可视化结果和浏览器console打印结果分别截图并粘贴至对应报告中。

## 模块F：综合分析（10分）

### 任务一：如何解决Job运行效率低的问题

在模块B中出现某些Job运行时间较长，你认为可能是哪些情况造成？有什么相应的处理办法吗？将内容编写至对应报告中。

### 任务二：如何解决数据倾斜问题

当使用spark产生了数据倾斜的情况时,请问有哪些方法可以解决这个问题呢？将内容编写至对应报告中。

### 任务三：简要描述任务过程中的问题并进行总结

将内容编写至对应报告中。