# 指标评估操作文档

# 指标类型

1.评估指标有两大类（详见《性能评估指标方案.pptx》）

* 1. 分类指标：包括查全率，查准率，F1值，ROC，AUC
  2. 排序指标：包括topK准确率，平均准确率，ndcg，recall@K

# 整理正样本数据

1. 如果有新的正样本，则需要先导入到hive库的”**同行正样本表**”（tb\_acc\_sample），

如果没有新的正样本则忽略此步骤。

数据表说明：

2.1同行正样本表（tb\_acc\_sample）：

|  |  |  |
| --- | --- | --- |
| 列名 | 类型 | 字段说明 |
| id | string | 样本Id （编号=名称+时间）如：10月5号花都的数据id可用 huadu\_1005 |
| obja | string | 目标a（车牌，imsi,phone等） |
| objb | string | 目标b（车牌，imsi,phone等） |
| label | int | 标签（正类1，负类0） |
| type | string | 算法：person-person/car-car/phone-phone/imei-imei/imsi-imsi/mac-mac/phone-imei/phone-mac/imei-imsi/imsi-mac/person-phone/person-imsi/person-imei/person-mac/car-phone/car-imsi/car-imei/car-mac |
| bd | string | 样本发生时间20201005 |

2.2 如果没有正样本，从hive表dm\_lcss 取相应的top几进行统计

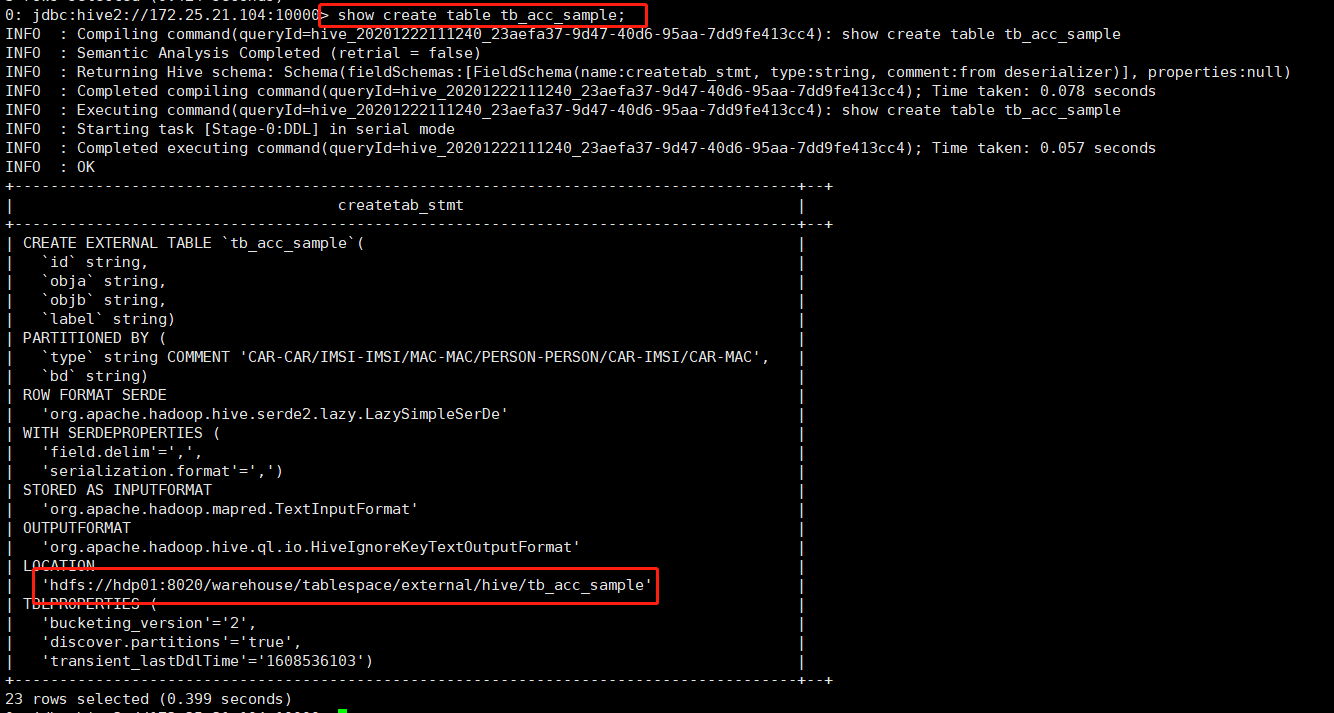
|  |
| --- |
| 1. 进入beeline ,执行show partition dm\_lcss;获得【最新分区】  2.将【最新分区】替换下面sql里的20201109字段，并再beeline 里执行  insert into table tb\_acc\_sample partition(type,bd)  select concat(type,stat\_date) as id, object\_id\_a as obja  , object\_id\_b as objb, 1 as label ,type ,stat\_date as bd  from (  select object\_id\_a,object\_id\_b,type ,stat\_date  ,row\_number() over(partition by type order by sim\_score desc ) as rn  from dm\_lcss where stat\_date=20201109  )where rn <=100; |

进入beeline客户端，查看是否生成成功。命令【select \* from tb\_acc\_sample where bd=最新分区】，如果有数据则成功，跳过下面步骤直接执行[指标相关图表生成](#_指标相关图表生成)，没有数据则检查是否dm\_lcss对应最新分区没有数据。

如果有正样本：将正样本按上面正样本表中的字段顺序整理，以逗号分隔成如下格式，并以utf-8的格式保存文件。

|  |
| --- |
| huadu\_1105,粤LM590P,粤AD78255,1,car-car,20201105  huadu\_1105,沪XX590P,浙YY78255,1,car-car,20201105 |

2.3查看正样本路径，命令【进入beeline客户端，show create table tb\_acc\_sample;】



然后根据type类型和正样本的日期创建目录，命令是：【hdfs dfs -mkdir hdfs://xxxx（根据实际情况更改）/tb\_acc\_sample/type=imsi-imsi（根据实际情况更改）/bd=20201105（根据实际情况更改）】

并将步骤1.2中保存的文件上传到红框中的目录下。命令【hdfs dfs -put 步骤1.2文件本地目录(xxx/xxxx) 红框中的目录“hdfs://xxxxx/tb\_acc\_sample/” 】

2.4进入beeline客户端，查看是否上传文件成功。命令【select \* from tb\_acc\_sample where id=”步骤1.2的样本id”】

# 指标相关图表生成

3.1根据情况配置执行json参数，参数解释如下表所示：

|  |  |  |  |
| --- | --- | --- | --- |
| cnName | enName | Value | desc |
| 算法名 | Algorithm | lcss | 算法名：五个算法：lcss/fptree  /dtw/entropy/fusion |
| 类型 | aType | car-car | 类型：car-car/imsi-imsi/mac-mac/person-person/car-imsi/car-mac |
| 评估类型 | evaluationType | rank | 评估类型:rank |
| 预测数据分区时间 | statDate | 20201112 | 预测数据分区时间。结果表dm\_relation的分区时间，一般用最大分区，具体可根据实际情况定时间，命令（show partitions dm\_relation;  ） |
| 正样本集合ID | posId | huadu\_1105 | 正样本的ID |

3.2登录xxl-job调度数据库，按照3.1更改executor\_param字段中的json参数，并执行。

|  |
| --- |
| update pd\_job.xxl\_job\_qrtz\_trigger\_info  set executor\_param='{"batchId":0,"analyzeModel":{"env":{"shellName":"data-algorithm.sh","deployEnvName":"ALG\_DEPLOY\_HOME"},"params":[{"cnName":"算法名","enName":"algorithm","desc":"算法名","value":"fusion"},{"cnName":"类型","enName":"aType","desc":"类型","value":"car-car"},{"cnName":"数据库名称","enName":"dataBaseName","desc":"数据库名称","value":"default"},{"cnName":"数据库类型","enName":"databaseType","desc":"数据库类型","value":"hive"},{"cnName":"评估类型","enName":"evaluationType","desc":"评估类型","value":"rank"},{"cnName":"statDate","enName":"statDate","desc":"statDate","value":"20210110"},{"cnName":"posId","enName":"posId","desc":"posId","value":"lcss\_car\_car\_0901\_12"}],"isTag":1,"tagInfo":[],"mainClass":"com.suntek.algorithm.evaluation.AcompanyEvaluation","modelId":0,"modelName":"模型评估","stepId":99401,"taskId":99401,"descInfo":""}}'  where id='99401';  update pd\_job.ttask\_step\_instance set status=3 where taskId='99401';  commit; |

3.3 执行脚本，将下面的命令更改参数后再linux上执行。

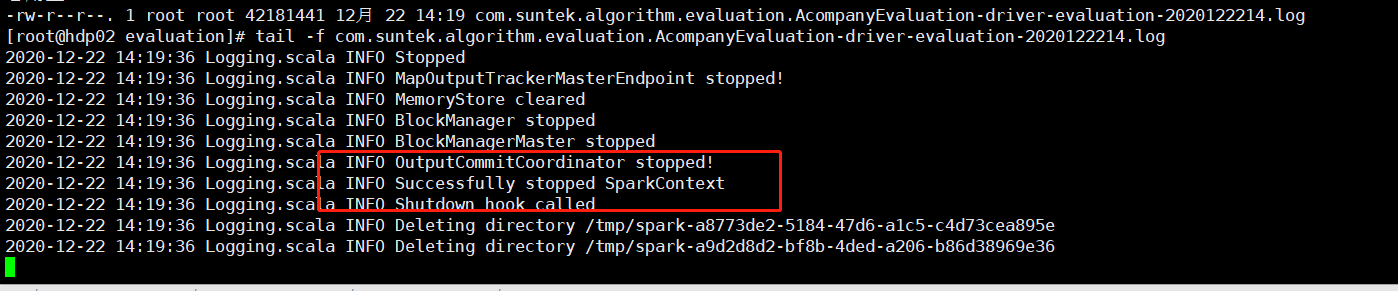
3.3.1将下面的 ip端口改为调度具体的ip和端口

3.3.2 batchId为样本发生时间,是正样本表的时间bd

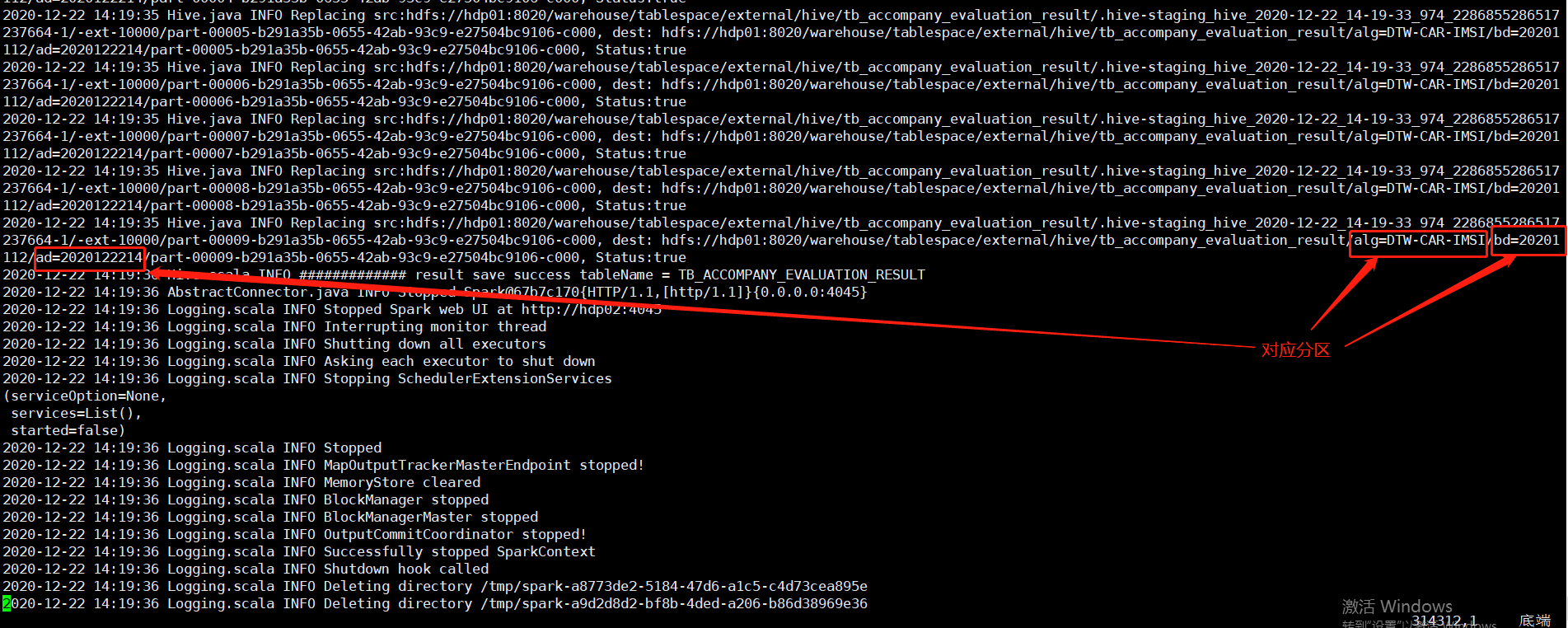
3.3.3 jobId为99401

|  |
| --- |
| curl -X GET "http://IP:端口/das/service/manage/task/taskSingleBatchRun?batchId=20200907000000&jobId=99401" -H "accept: \*/\*" |

3.4 进入目录 /data/data-algorithm/logs/evaluation/查看日志。 如果日志末尾看到下图所示则执行成功，如果不是需重新检查原因。



3.5登录beeline查询 评估结果表（TB\_ACCOMPANY\_EVALUATION\_RESULT）对应分区是否有数据。命令【select \* from tb\_accompany\_evaluation\_result where alg='算法名' and bd=时间1 and ad=时间2】。可以使用命令进行导出（beeline --outputformat=csv2 -e "select \* from tb\_accompany\_evaluation\_result where alg='算法名' and bd=时间1 and ad=时间2" > 输出文件.sql）



|  |  |  |
| --- | --- | --- |
| 列名 | 字段类型 | 字段说明 |
| posid | string | 正样本id |
| negid | string | 负样本id |
| thr\_or\_fpr | string | 如果evaluation字段为查准率，召回率,f1值，则thr\_or\_fpr是阈值，val\_or\_tpr是查准率，召回率,f1值 如果evaluation字段为ROC则thr\_or\_fpr为假阳性率，val\_or\_tpr为真阳性率，如果evaluation字段为top准确率，ndcg，则thr\_or\_fpr为topk（1，...20），val\_or\_tpr是top准确率，ndcg |
| val\_or\_tpr | string |
| evaluation | string | 查准率，召回率,f1值，ROC，AUC，topK准确率，平均准确率（MAP），ndcg |
| alg | string | fptree-car-imsi,fptree-car-car,fptree-imsi-imsi,lcss-car-imsi,lcss-car-car,lcss-imsi-imsi |
| bd | string | 样本发生的时间20201005 |
| ad | string | 评估指标计算时间2020111711 |