

**环境要求：**

Hadoop 2.7 + Spark 2.3

Python 3.6

**Python库：**

pyspark、kafka、pandas、scipy、sklearn、tsfel

**模块名：读Kafka**

依赖库：pyspark、kafka

功能：Kafka消费者，舍弃4分钟前的数据。如果读到速率异常告警数据，则获取速率异常时间戳，4分钟后，获得异常时间点前后4分钟的实时解码参数数据，启动后续模块。如果在8分钟范围内读到多个速率异常告警，则并行处理后续模块。

输入：json格式待定

输出：速率异常时间点、异常时间点前后4分钟内带有时间戳的实时解码参数数据

输入数据样例：

{

"uuid": "d85f8a8525cc443a9f4a56d1789da477",

"m\_uuid": "c207a6e5a206431f8ff2eaf998b937c9",

"device\_sn": "ccb33f35-6369-4f21-8491-bb995d538639",

"num": 6,

"timesec": 1653483510,

"testtime": "1970-01-20 11:18:03",

"source": 53100,

"dest": 445,

"protocol": 6,

"flow\_bytes": 72,

"device\_id": 184,

"scene\_id": 2,

"dt": "19700120",

"dt\_hour": "1970012011",

"dt\_min": "197001201118",

"dt\_sec": "19700120111803",

"p\_scene\_id": 1

}

**模块名：时序数据采样**

Python脚本文件：aidiag\_ts\_sample.py

依赖库：pandas

功能：将带有时间戳的实时解码参数数据整理成按秒采样的参数数据，范围为异常时间点前后4分钟。如果该秒没有数据则留空；如果该秒有多个数据则取平均值。该模块的输出可以保存到数据库作为模型离线训练的数据集。

接口名：def ts\_sample(df\_raw: DataFrame) -> DataFrame（pandas.core.frame.DataFrame）

输入：速率异常时间点、异常时间点前后4分钟内带有时间戳的实时解码参数数据

输出：速率异常时间点、异常时间点前后4分钟内按秒采样的参数数据（DataFrame对象）

时序数据字段：

sec,SS\_RSRP,SS\_RSRQ,SS\_SINR,PUCCH\_TxPower,PUSCH\_TxPower,Pathloss,MCSAvg\_DL,MCSBest\_DL,MCSMost\_DL,MCSAvg\_UL,MCSBest\_UL,MCSMost\_UL,CQI\_Avg,CQI\_Best,CQI\_Most,PDSCH\_BLER,PDSCH\_iBLER,PDSCH\_rBLER,PUSCH\_BLER,PUSCH\_iBLER,PUSCH\_rBLER

其中sec为按秒采样的时间

DataFrame固定为480行（8分钟）

**模块名：Pandas数据处理**

Python脚本文件：aidiag\_ts\_transform.py

依赖库：pandas、scipy

功能：对按秒采样的参数数据进行进一步数据转换处理，如线性插值

接口名：def ts\_transform(df\_param: DataFrame) -> DataFrame

输入：速率异常时间点、异常时间点前后4分钟内按秒采样的参数数据（DataFrame对象）

输出：转换后的参数数据（DataFrame对象）

**模块名：tsfel时序特征提取**

Python脚本文件：aidiag\_ts\_feature.py

依赖库：tsfel、pandas

功能：从转换后的参数数据中，提取统计特征与时序特征。该模块代码和离线训练时的代码相同，可以复用。

接口名：def ts\_feature(df\_data: DataFrame) -> numpy.ndarray

输入：转换后的参数数据（DataFrame对象）

输出：时序特征（numpy矩阵）

**模块名：模型预测**

Python脚本文件：aidiag\_predict.py

依赖库：sklearn

功能：将时序特征输入已经训练好的模型，得到预测的故障原因

接口名：def predict(feature: numpy.ndarray, model: model) -> dict

输入：时序特征（numpy矩阵）、已经离线训练好的模型（model文件）

输出：模型预测的故障原因概率分布（覆盖/干扰/其他）（dict）

输出数据样例：

{

"predict\_reason": “弱覆盖”,

"predict\_reason\_prob": {

“弱覆盖”: 0.6,

“干扰”: 0.3,

“其他”: 0.1,

}

}

**模块名：写Kafka**

依赖库：pyspark、kafka

功能：Kafka生产者，将模型预测的故障原因与相应的异常时间点一起写入消息队列。

输入：速率异常时间点、模型预测的故障原因（覆盖/干扰/其他）

输出：json格式待定

输出数据样例：

{

"uuid": "d85f8a8525cc443a9f4a56d1789da477",（与异常告警发生信息一致）

"m\_uuid": "c207a6e5a206431f8ff2eaf998b937c9",（与异常告警发生信息一致）

"device\_sn": "ccb33f35-6369-4f21-8491-bb995d538639",（与异常告警发生信息一致）

"dt": "19700120",

"dt\_hour": "1970012011",

"dt\_min": "197001201118",

"dt\_sec": "19700120111803",（与异常告警发生信息一致）

"predict\_reason": “弱覆盖”, （模型预测的故障原因）

"predict\_reason\_prob": { （模型预测的故障原因概率分布）

“弱覆盖”: 0.6,

“干扰”: 0.3,

“其他”: 0.1,

}

}