Web进程链恶意命令检测算法功能与使用说明书

Catalog 目 录

1 算法功能说明 3

1.1 恶意命令检测 3

1.2 告警抖动控制 3

2 算法接口 3

2.1 接口原型 3

2.1.1 训练接口 3

2.1.2 预测接口 3

2.2 功能描述 3

2.3 接口类型 4

2.4 输入 4

2.5 返回值 4

2.6 参数的数据类型定义 4

2.6.1 数据结构 4

2.7 其它约束 4

3 安装指导 5

3.1 环境要求 5

3.2 端口矩阵 5

3.3 部署说明 6

3.4 集成样例 6

# 算法功能说明

## 恶意命令检测

通过n-gram-knn算法训练出检测模型，实现对线上业务web进程链中的指令的检测过程，判断指令是否恶意。

## 告警抖动控制

新业务出现时，对告警进行控制，判断有新业务出现时，对来源进行记录，恶意指令不再告警。

# 算法接口

## 接口原型

算法包括训练与预测两部分

### 训练接口

train\_df(sc, model\_dir, normal\_data\_dir)

### 预测接口

predict(sc, model\_dir)

## 功能描述

Train\_df接口接收三个参数，sc表示当前sparkContext上下文，model\_dir表示模型最终的输出路径，normal\_data\_dir表示模型读取数据的路径。函数会从normal\_data\_dir中读取日志文件，训练出模型后，保存至model\_dir中

Predict接收两个参数，sc表示当前sparkContext上下文，model\_dir表示加载模型的路径，函数使用spark streaming流处理数据，收到Dstream数据后，转换为对应的Dataframe，随后根据指令的来源加载模型，使用对应模型预测该指令是否恶意。

## 接口类型

python函数

## 输入

|  |  |  |
| --- | --- | --- |
| **参数名称** | **参数类型** | **说明** |
| Model\_dir | Str | 保存或加载模型的路径 |
| Normal\_data\_dir | str | 加载训练数据的路径 |
| Sc | SparkContext | SparkContext上下文 |

## 返回值

|  |  |
| --- | --- |
| **返回值类型** | **含义** |
| None |  |

## 参数的数据类型定义

无

### 数据结构

无

## 其它约束

版本限制：spark 2.3.2 python3

# 安装指导

## 环境要求

Spark/2.3.2 python/3

## 端口矩阵

暂无

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** |  | **端口** | **说明** |
| 1 |  | 1389 |  |

## 部署说明

* **将算法代码文件上传至mrs的master节点**
* **使用conda创建虚拟环境安装需要的python依赖**
* **打包虚拟环境**
* **使用spark-submit提交python任务**

*注：当前没有独立部署，待有独立部署需求时，需要与工具确认后给出具体的命令。*

## 集成样例

测试数据格式为json格式的文本，其字段说明如下

|  |  |
| --- | --- |
| CreateTime | 进程创建时间 |
| ppid | 父进程id |
| pname | 父进程名称 |
| pcmdline | 父进程命令 |
| pexe | 父进程执行路径 |
| cpid | 子进程id |
| cname | 子进程名称 |
| ccmdline | 子进程命令 |
| cexe | 子进程执行路径 |

示例数据

**{**

**"CreateTime"**: 1587970622016**,**

**"ppid"**: 131359**,**

**"pname"**: "\_checkTomcat"**,**

**"pcmdline"**: "/bin/bash/opt/webserver/programs/bin/\_checkTomcattomcat\_cas"**,**

**"pexe"**: "/bin/bash"**,**

**"cpid"**: 131369**,**

**"cname"**: "\_checkTomcat"**,**

**"ccmdline"**: "/bin/bash/opt/webserver/programs/bin/\_checkTomcattomcat\_cas"**,**

**"cexe"**: "/bin/bash"

**}**

调用样例

*if* args.stage == 'train':  
 train\_df(sc, args.model\_path, args.normal\_data\_dir)  
*elif* args.stage == 'predict':  
 predict(sc, 'data')