

# 腾讯云 TI 平台 TI-ONE 快速入门 产品文档





【版权声明】

©2013-2022 腾讯云版权所有

本文档(含所有文字、数据、图片等内容)完整的著作权归腾讯云计算(北京)有限责任公司单独所有,未经腾讯 云事先明确书面许可,任何主体不得以任何形式复制、修改、使用、抄袭、传播本文档全部或部分内容。前述行为 构成对腾讯云著作权的侵犯,腾讯云将依法采取措施追究法律责任。

【商标声明】

# 🔗 腾讯云

及其它腾讯云服务相关的商标均为腾讯云计算(北京)有限责任公司及其关联公司所有。本文档涉及的第三方主体 的商标,依法由权利人所有。未经腾讯云及有关权利人书面许可,任何主体不得以任何方式对前述商标进行使用、 复制、修改、传播、抄录等行为,否则将构成对腾讯云及有关权利人商标权的侵犯,腾讯云将依法采取措施追究法 律责任。

【服务声明】

本文档意在向您介绍腾讯云全部或部分产品、服务的当时的相关概况,部分产品、服务的内容可能不时有所调整。 您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定,除非双方另有约定,否 则,腾讯云对本文档内容不做任何明示或默示的承诺或保证。

【联系我们】

我们致力于为您提供个性化的售前购买咨询服务,及相应的技术售后服务,任何问题请联系 4009100100。



# 文档目录

快速入门

注册与开通服务 使用可视化建模构建模型 使用 Notebook 构建模型 使用 TI SDK 构建模型



# 快速入门 注册与开通服务

最近更新时间: 2022-02-16 15:05:24

为了方便您有效地使用腾讯云 TI 平台 TI-ONE,请先完成以下步骤:

### 步骤1: 注册腾讯云账号

若您已在腾讯云注册,可忽略此步骤。

点此注册腾讯云账号

# 步骤2: 开通腾讯云 TI 平台 TI-ONE 服务

您在使用腾讯云 TI 平台 TI-ONE 前,需要先开通所需地区的后付费计费模式才可正常使用。详情请参考 购<mark>买指</mark> <mark>南</mark>。

点此进入开通服务页面

#### 步骤3:一键授权

进入 腾讯云 TI 平台 TI-ONE 控制台,在弹出的页面上开通一键授权。

1. 单击前往一键授权, 页面将跳转至访问管理控制台。

2. 单击**同意授权**,即可创建服务预设角色并授予腾讯云 TI 平台 TI-ONE 相关权限。

### 步骤4: 创建 COS 存储桶

完成**一键授权**后,在此步骤,您可选择创建个人的付费 COS 存储桶。

・ 创建个人付费 COS 存储桶

您可以进入 COS 控制台,在存储桶列表页面创建存储桶,详情请参考 创建存储桶。 创建好的存储桶将用于平台任务数据的存放,包括工程任务、Notebook 任务、SDK 任务。

#### △ 注意:

创建 COS 存储桶时,所属地域需跟平台开通服务的地域一致,如平台地域处于广州,创建存储桶时所属地 域应选择广州。





#### 注意事项

- 1. 一键授权只需要操作一次,授权后则访问管理中已经自动创建角色。下一次新建工程时直接选择存储桶即可。
- 2. 用户首次使用子账号登录时,需要主账号先给子账号授予 QcloudCamRoleFullAccess 权限,再单击一键授 权即可按上述步骤完成授权。

主账号添加策略方法如下:

- 在主账号的 用户列表 中找到此子账号,单击 授权。
- 在策略列表中输入"QcloudCamRoleFullAccess",选择相应的策略名并单击确定。
   关联策略

<b>策略列表</b> (共1条)			ī	3选择(1条)	
QcloudCamRoleFullAccess	0	Q		策略名	策略类型
策略名	策略类型 ▼			QcloudCamRoleFullAccess	预设策略
✓ QcloudCamRoleFullAccess 用户与权限(CAM)角色全读写访问权限	预设策略			用户与权限(CAM)角色全读写访问权限	
			$\Leftrightarrow$		
±₩約/±₅ы###₩/二々)#					
又讨如工知顺建进行多匹					
		确定		取消	

3. 腾讯云 TI 平台支持主账号,子账号,协作者身份登录 TIONE 控制台,详情请参考 腾讯云用户类型。



# 使用可视化建模构建模型

最近更新时间: 2022-01-10 15:31:49

### 操作背景

- 腾讯云 TI 平台 TI-ONE 通过可视化的拖拽布局,组合各种数据源、组件、算法、模型及评估模块,为 AI 工程 师打造从数据预处理、模型训练、到模型评估的全流程开发支持。更多详情请见 可视化建模简介。
- 为了方便您有效地使用腾讯云 TI 平台 TI-ONE 的可视化建模,本文档将通过一个乳腺癌预测案例向您演示使用流程。
- 在使用之前,请确保您已经完成了 注册与开通服务。

#### 操作步骤

#### 步骤1:新建工程与任务流

使用腾讯云 TI 平台 TI-ONE 建模,首先需要完成新建工程和任务流。

#### 新建工程

- 登录 腾讯云 TI 平台 TI−ONE 控制台,将平台地域切换为您开通服务时所选地域。在工程列表页面,单击**我的** 工程>新建工程。
- 根据提示填写工程名称、工程描述、COS Bucket。请在下拉列表处选择 Bucket 名称,该工程里的训练数据、中间结果等内容将存入此 Bucket。注意 COS Bucket 需要与平台处于相同区域,若无 COS Bucket 可选,请前往 COS 控制台 新建,详情请参考 创建存储桶。

#### 新建任务流

在工程中单击"+"号,您可新建自定义工作流。创建完成后单击进入画布。

#### 步骤2:数据接入

腾讯云 TI 平台 TI-ONE 提供两种数据源途径:本地数据和 COS 数据集。

#### 数据集介绍

本次任务我们采用公开的 乳腺癌数据集,该数据集共包含569个样本,其中357个阳性(y = 1)样本,212个阴性 (y = 0)样本;每个样本有32个特征,但本次实验中选取其中10个特征。

数据集具体字段信息如下(表格中包括每列字段的名称和对应取值范围):

特征和标签(Attribute)	取值范围(domain)
Clump Thicknedd	$1 \le n \le 10$



特征和标签(Attribute)	取值范围(domain)
Uniformity of Cell Size	$1 \le n \le 10$
Uniformity of Cell Shape	$1 \le n \le 10$
Marginal Adhsion	$1 \le n \le 10$
Single Epithelial Cell Size	$1 \le n \le 10$
Bare Nuclei	$1 \le n \le 10$
Bland Chromation	$1 \le n \le 10$
Normal Nucleoli	$1 \le n \le 10$
Mitoses	$1 \le n \le 10$
(标签y)Class	(2 for begin, 4 for maligant)

**数据集具体内容抽样展示如下**(前9列是特征取值,最后1列是标签取值):

1	3230021011
2	4130101410
3	4170110111
4	3270021100
5	3261010211
6	4251111000
7	4180120010
8	3220110010
9	3200111310
10	3185011000
11	4250110110
12	5130111010
13	4160101410
14	4150111010
15	3250110101
16	2240120410
17	4221101010

#### 数据导入

- 1. 为方便用户操作,我们直接提供本实验所需的数据集 cancer.csv,请直接下载并保存到本地。
- 2. 在腾讯云 TI 平台控制台的左侧导航栏,选择输入>数据源>\*\* 本地数据\*\*,并拖入画布中。
- 3. 在**本地数据**对应右边弹出的设置窗口中选择**算法 IO 参数**中的"数据文件"上传,单击**数据文件**输入框,选择刚 保存到本地的 cancer.csv 数据集上传,单击确定。





#### 步骤3:数据预处理

在工程实践中,为了便于数据建模,需要对数据集中存在的缺失、重复等问题进行预处理,同时需要将字符串转换 为可以参与建模的数值形式。

在模型训练时我们通常还会对数据进行切分,分为训练集 training set、验证集 validation set、测试集 test set。训练集用来训练模型,验证集用于调节模型超参数,测试集用来整体评估模型性能。 本实验将数据切割成训 练集和测试集,比例为 8 : 2 ( 80%用于训练,20%用于测试 ) 。

1. 在控制台的左侧导航栏,选择算法>机器学习算法>数据预处理>数据切分,并拖入画布中。

2. 将本地数据的输出桩连接到数据切分的输入桩。

- 3. 填写参数:
  - 。 算法 IO 参数 \*\*中的各个数据路径参数根据连线自动生成,无需更改,您可单击高级设置\*\*查看
  - 。 输入文件类型: csv
  - 。 输入数据包含 header 信息: 否
  - 。 输入数据分割符: 空格
  - 。 切分比例: 0.8 (第一个输出桩为80%的训练数据,第二个输出桩为20%的测试数据)
  - 。 driver 节点资源类型: 您可按需选择。
  - executor 节点资源类型:您可按需选择。
  - 。 其余参数可默认





#### 步骤4: 模型训练

腾讯云 TI 平台 TI-ONE 内置百余种机器学习与深度学习算子,您可以在左侧算法栏中选择合适的算子进行模型搭 建。找到该算子拖入画布中,您可以根据节点图式进行连线,过程中任何连线或算子选择有误都可以通过右键删 除。

#### 1. 选择算子

- 在控制台的左侧导航栏,选择算法>机器学习算法>分类>决策树分类,并拖入画布中。
- 将数据切分左边的输出桩连接到决策树分类左边的输入桩,表示训练数据输入模型;右边的输出桩连接到决策树
   分类最左侧的蓝色小烧杯处,表示将测试数据输入模型。

#### 2. 配置算法参数

单击该算子,在右侧弹框中填写**决策树分类**相关参数。您可按需填写,本案例参数配置如下:

- 特征列: 0-8。
- 标签列: 9。
- 输入数据包含 header 信息:是
- 输入数据分隔符: 逗号
- 模型保存格式: ML
- driver 节点资源类型:您可按需选择。
- executor 节点资源类型:您可按需选择。
- 其余参数可默认

#### 3. 配置模型参数



左侧蓝色小烧杯代表模型,单击填写模型参数。您可按需填写,本案例参数配置如下:

- 模型更新方式: 手动运行
- 模型运行方式: 自动运行
- 特征列: 0-8
- 输入/输出数据包含 header 信息:是
- 输入/输出数据分隔符: 逗号
- 模型导入格式: ML
- driver 节点资源类型:您可按需选择。
- executor 节点资源类型:您可按需选择。



#### 步骤5: 模型评估

至此,我们已经利用 TI-ONE 搭建了乳腺癌预测模型,那么我们训练出的模型效果如何呢?本文中,我们可以利用 腾讯云 TI 平台的**模型评估**算子**二分类任务评估**来查看模型训练效果。

1. 在腾讯云 TI 平台 TI-ONE 控制台的左侧导航栏,选择输出>模型评估>二分类任务评估,并拖入画布。



#### 2. 单击二分类任务评估,填写参数:

- 。标签列:9
- 。 是否是打分项:否
- 。 输入数据是否包含 header 信息:是
- 。 输入数据分隔符: 逗号
- 。 预测列:10
- 。 driver 节点资源类型:您可按需选择。
- 。 executor 节点资源类型:您可按需选择。
- 。 其余参数可默认



#### 步骤6:运行工作流

单击画布上方运行可运行工作流。

#### 步骤7: 查看模型效果



#### 运行成功后,右键选择**二分类任务评估>评估指标**,即可查看模型效果。

评估指标



metricName 🗘	metric 🗢
precision	0.7142857142857143
recall	0.29411764705882354
fl	0.4166666666666666666

至此,我们完成了使用腾讯云 TI 平台 TI-ONE 的可视化建模训练模型的流程。



# 使用 Notebook 构建模型

最近更新时间: 2021-12-29 16:13:43

### 操作背景

- Notebook 是腾讯云 TI 平台 TI-ONE 为开发者量身打造的灵活的交互式开发工具,您可以在腾讯云 TI 平台 Notebook 中完成数据准备、数据预处理、算法调试与模型训练,无需多平台切换,更多详情请见 Notebook 简介。
- 为了方便您有效地使用腾讯云 TI 平台 TI-ONE 的 Notebook,本文档将通过一个案例向您演示使用流程。
- 在使用之前,请确保您已经完成了 注册与开通服务。

### 操作步骤

#### 步骤1: 创建实例

- 1. 登录 腾讯云 TI 平台 TI-ONE 控制台 ,单击左侧导航栏的Notebook,页面将跳转至 Notebook 的实例列表 页面,此页面将罗列用户创建的所有 Notebook 实例。
- 2. 在 Notebook 实例列表页,单击左上角新增实例,跳转至创建 Notebook 实例的设置页面。
- 3. 在**新增实例**页面,可填写以下相关字段:
  - 。 **地区**:此字段不可修改,将自动显示平台选择的地区。
  - 。 Notebook 名称: 设置此 Notebook 实例的名称。
  - 资源选择:选择此实例需要配置的资源。(注意:只要 Notebook 实例处于运行中,都将对配置的资源进行 按时收费。)
  - 存储大小: Notebook 实例的存储大小(以 GB 为单位),最小值为10GB且为10的倍数,注意:请大于当前硬盘值,最大值为16380GB(约16TB)。
  - 。 单击**高级设置**此选项可根据需求进行设置(默认不展开),可配置:
    - Root 权限:选择是否赋予 root 权限来访问 Notebook。如果启用 Root 权限,则所有 Notebook 实例用户具有管理员权限,并且可以访问和编辑实例上的所有文件。
    - **生命周期配置:**选择是否使用生命周期脚本。
    - Git 存储: 此为可选项,用户可以前往 Git 存储库-新增存储库进行配置。
    - VPC: 用户可以选择配置自有的 VPC 网络。
    - CLS 日志服务: 用户可以自行选择是否开通 CLS 日志服务。
    - 自动停止:开启该选项后,该实例将在运行时长超过您选择的时长后自动停止。
  - 。 且平台会根据您以上选择的配置,计算对应的价格:
    - 计算资源价格:平台根据您选择的配置显示相关价格。
    - 存储资源价格:平台根据您选择的配置显示相关价格。
    - 总价:平台根据您选择的配置显示相关价格。



- 以上信息已填写完成后,单击创建,Notebook 列表中将新增一条实例记录,用户可单击状态查看实例创建进程,当实例状态由创建中变为运行中时,单击打开进入 Notebook 实例内部。
- 5. 进入实例内部后,您可以根据需要设置内核环境。本案例使用 conda\_tensorflow\_py3。



#### 步骤2:数据导入

本案例代码来自 Tensorflow 官方项目。我们使用公共的鸢尾花(iris)数据集训练模型,该数据集包含四个特征,分别是花萼长度、花萼宽度、花瓣长度、花瓣宽度,我们根据这四个特征将鸢尾花分成三种物种。

```
CSV_COLUMN_NAMES = ['SepalLength', 'SepalWidth', 'PetalLength', 'PetalWidth', 'Species']
SPECIES = ['Setosa', 'Versicolor', 'Virginica']
```

本测试数据存储在 COS 广州地域,您可 单击下载并查看,在 Notebook 中导入所需数据。

!pip install wget
import wget, tarfile
filename = wget.download("https://tesla-ap-guangzhou-1256322946.cos.ap-guangzhou.myqclo
ud.com/cephfs/tesla\_common/deeplearning/dataset/contest/demo.zip")
print(filename)

import zipfile
zFile = zipfile.ZipFile(filename, "r")



for fileM in zFile.namelist():
zFile.extract(fileM, "./")
print(fileM)
zFile.close();

#### 步骤3:模型训练

您可以自行编写代码进行模型构建、模型训练、模型评估。

import pandas as pd import tensorflow as tf

CSV\_COLUMN\_NAMES = ['SepalLength', 'SepalWidth', 'PetalLength', 'PetalWidth', 'Species'] SPECIES = ['Setosa', 'Versicolor', 'Virginica']

def load\_train\_data(train\_path, y\_name='Species'):
"""Returns the iris dataset as (train x, train y)"""

train = pd.read\_csv(train\_path, names=CSV\_COLUMN\_NAMES, header=0)
train\_x, train\_y = train, train.pop(y\_name)

return (train\_x, train\_y)

def load\_test\_data(test\_path):
"""Returns the iris dataset as test x"""

test = pd.read\_csv(test\_path, names=CSV\_COLUMN\_NAMES[:-1], header=0)

return test

def train\_input\_fn(features, labels, batch\_size):
"""An input function for training"""
# Convert the inputs to a Dataset.





dataset = tf.data.Dataset.from\_tensor\_slices((dict(features), labels))

# Shuffle, repeat, and batch the examples.
dataset = dataset.shuffle(1000).repeat().batch(batch\_size)

# Return the dataset. return dataset

def eval\_input\_fn(features, labels, batch\_size):
"""An input function for evaluation or prediction"""
features=dict(features)
if labels is None:
# No labels, use only features.
inputs = features
else:
inputs = (features, labels)

# Convert the inputs to a Dataset.
dataset = tf.data.Dataset.from\_tensor\_slices(inputs)

# Batch the examples
assert batch\_size is not None, "batch\_size must not be None"
dataset = dataset.batch(batch\_size)

# Return the dataset

return dataset

import os
def main():

 $batch_size = 100$ 

train\_steps = 1000

# Fetch the data

(train\_x, train\_y) = load\_train\_data("iris\_training.csv")



# Feature columns describe how to use the input.

my\_feature\_columns = []
for key in train\_x.keys():
my\_feature\_columns.append(tf.feature\_column.numeric\_column(key=key))

# Build 2 hidden layer DNN with 10, 10 units respectively.

classifier = tf.estimator.DNNClassifier( feature\_columns=my\_feature\_columns, # Two hidden layers of 10 nodes each. hidden\_units=[10, 10], # The model must choose between 3 classes. n\_classes=3, )

# Train the Model.

classifier.train( input\_fn=lambda: train\_input\_fn(train\_x, train\_y, batch\_size), steps=train\_steps)

# Generate predictions from the model
test\_x = load\_test\_data("iris\_test.csv")

```
predictions = classifier.predict(
input_fn=lambda: eval_input_fn(test_x,
labels=None,
batch_size=batch_size))
result = []
template = 'Prediction is "{}" ({:.1f}%)'
for pred_dict in predictions:
class_id = pred_dict['class_ids'][0]
probability = pred_dict['probabilities'][class_id]
print(template.format(SPECIES[class_id],
100 * probability))
result.append(class_id)
```

result\_df = pd.DataFrame(data=result)



result\_df.to\_csv("result\_file", index=False)

print("result file is saved")

#### main()

步骤4:结果保存

#### 1. 结果文件路径

您可以自行指定将实验结果保存到 个人存储桶 中的特定 COS 文件路径下。

path:结果文件路径。

bucket:指定存储桶。注意:请指定用户对应地域下的个人 COS 存储桶,使用示例中的存储桶会导致报错。 key\_prefix:存储桶下 COS 路径地址。

## from ti import session ti\_session = session.Session() inputs = ti\_session.upload\_data(path="result\_file", bucket="demo-project-ap-guangzhou-1259 675134", key\_prefix="contest")

#### 2. 结果文件查看

您可以到 COS 中您指定的路径下查看结果文件。此外,您可以自行下载文件,单击**详情**,还可在详情页面获取**对 象地址**。

← 返回桶列表	yonghao-1259675134 / contest_result					文档排
文件列表	<b>上传文件 创建文件夹</b> 更多操作	¥			请输入前缀	Q, A
基础配置	文件名	大小	存储类型	更新时间	操作	
高级配置	result.csv	62B	标准存储	2020-03-19 17:15:49	下载 详情	检索删除
域名管理						
图片处理						

至此,我们完成了使用腾讯云 TI 平台 TI-ONE 的 Notebook 训练模型的流程。



# 使用 TI SDK 构建模型

最近更新时间: 2021-12-22 11:55:02

### 操作背景

- TI SDK 是腾讯云 TI 平台 TI-ONE 提供的开源软件包。用户可以使用 TI SDK 提交机器学习和深度学习训练 任务到 TI-ONE,更多详情请见 TI SDK 简介。
- 为了方便您有效地使用腾讯云 TI 平台 TI-ONE 的 TI SDK,本文档将通过一个分类案例向您演示使用流程。
- 在使用之前,请确保您已经完成了 注册与开通服务。

#### 操作步骤

#### 步骤1:数据说明

本案例训练数据已经存放在上海地域的 COS 中 单击查看,您可以下载到本地或者到 Notebook 中使用,在 Notebook 中使用须知:

- 请在上海地域执行例子中的代码
- 推荐使用 kernel: conda\_tensorflow\_py3
- 用户需在 Notebook 中同级目录下创建一个名为 code 的文件夹,然后将下载好的 main.py 文件上传到 code 文件夹中。

#### 步骤2: 引入依赖

from \_\_future\_\_ import absolute\_import, print\_function from ti import session import sys from ti.tensorflow import TensorFlow

#### 步骤3:初始化参数

#### # 初始化session

- ti\_session = session.Session()
- # 训练数据存放的路径.格式为cos://作为前缀拼接bucket名称再拼接训练数据存放的路径
- inputs = 'cos://tesla-ap-shanghai-1256322946/cephfs/tesla\_common/deeplearning/dataset/cont
  est/demo'
- # 指定训练集、测试集和训练结果路径

hyperparameters = {'train\_path': '/opt/ml/input/data/training/iris\_training.csv',



'test\_path': '/opt/ml/input/data/training/iris\_test.csv',
'result\_dir': '/opt/ml/output'}
# 授权给TI的服务角色
role = "TIONE\_QCSRole"

#### 步骤4:构建训练任务

# 创建一个Tensorflow Estimator tf\_estimator = TensorFlow(role=role, train\_instance\_count=1, train\_instance\_type='TI.3XLARGE24.12core24g', py\_version='py3', script\_mode=True, hyperparameters=hyperparameters, framework\_version='1.14.0', entry\_point='main.py', source\_dir='code')

#### 步骤5:提交任务,开始训练

# 提交Tensorflow训练任务 tf\_estimator.fit(inputs)

```
Training job request : {
```

"TrainingJobName": "tensorflow-2020-04-15-03-26-49-992",

```
"AlgorithmSpecification": {
```

"TrainingInputMode": "File",

"TrainingImageName": "ccr.ccs.tencentyun.com/ti\_public/tensorflow:1.14.0-py3"

},

"InputDataConfig": [

```
{
```

```
"DataSource": {
```

- "CosDataSource": {
- "DataDistributionType": "FullyReplicated",
- "DataType": "COSPrefix",



```
"Bucket": "tesla-ap-shanghai-1256322946",
"KeyPrefix": "cephfs/tesla common/deeplearning/dataset/contest/demo"
}
},
"ChannelName": "training"
],
"OutputDataConfig": {
"CosOutputBucket": "ti-ap-shanghai-100010875047-1259675134",
"CosOutputKeyPrefix": ""
},
"ResourceConfig": {
"InstanceCount": 1,
"InstanceType": "TI.3XLARGE24.12core24g",
"VolumeSizeInGB": 0
},
"RoleName": "TIONE QCSRole",
"StoppingCondition": {
"MaxRuntimeInSeconds": 86400
},
"HyperParameters": "{\"train_path\": \"\\\"/opt/ml/input/data/training/iris_training.csv\\\"\", \"tes
t_path\": \"\\\"/opt/ml/input/data/training/iris_test.csv\\\"\", \"result_dir\": \"\\\"/opt/ml/output\\\"\"
, \"ti_submit_directory\": \"\\\"cos://ti-ap-shanghai-100010875047-1259675134/tensorflow-2020
-04-15-03-26-49-992/source/source.tar.gz\\\"\", \"ti_program\": \"\\\"main.py\\\"\", \"ti_container_
log_level\": \"20\", \"ti_job_name\": \"\\\"tensorflow-2020-04-15-03-26-49-992\\\"\", \"ti_region\":
\"\\\"ap-shanghai\\\"\"}"
```

```
Training job response : {
```

"TrainingJobName": "tensorflow-2020-04-15-03-26-49-992", "Requestld": "06bb534b-f874-4ba6-9610-1d41f5044417"

5

#### 步骤6:查看输出模型

如果训练任务完成有模型输出,那么 TI 会将模型上传到 cos 上。

您可以通过 output\_path='cos://' 指定模型存储的 cos 路径。如果没指定,TI 会按以下格式创建存储桶 ti-[region]-[uin]-[appid] ,最终模型会放在 # {bucket}/# {job\_name}/output/ 目录下。



#### 步骤7: 查看训练日志

- TI 训练过程中会把日志上传到 腾讯云日志服务 中,使用腾讯云日志服务需要先进行开通,计费详情请参考 CLS 购买指南。
- TI 会针对训练任务创建 TiOne 日志集和 TrainingJob 日志主题,通过日志检索功能可以搜索对应训练任务的 日志。
- 目前 TI 会默认创建一个日志集(TI)和日志主题(TrainingJob)。
- TI 内置了任务名称(job)关键词,您可以通过以下条件过滤指定任务的日志:

#### job: #{job\_name}

更多日志检索语法请参考开启索引文档。

#### 步骤8: 查看提交历史和监控

您可以在 TI 产品控制台上的任务列表>SDK 任务中查看您提交的 SDK 任务历史,单击任务名称列中的任务名可 以看到任务的详细信息,在列表的右侧监控可以查看任务运行的资源监控图表。

至此,我们完成了使用腾讯云 TI 平台 TI-ONE 的 TI SDK 训练模型的流程。