阿里云机器学习

常见问题

常见问题

算法组件常见问题

目录

如果以上内容无法解决您的问题，请首先查看机器学习知识库，若问题仍得不到解决请粘贴logview（Tensorflow日志中的蓝色链接）到机器学习工单系统进行提问。

运行格式转换组件出错

格式转换组件默认起100个worker，请检查数据量是否大于100条。

机器学习平台数据展示出现“blob”字符

现象描述

在机器学习平台右键查看数据时部分文本变成“blob”字符。

解决方法

因为有部分字符不可转码，所以显示成为了“blob”，不影响下游节点的读取和处理。

x13-auto-arima 使用过程报错

x13-auto-arima的训练数据的规模有限制，不能超过1200条。

DOC2VEC 报错 CallExecutorToParseTaskFail

DOC2VEC支持的规模是（doc个数+word个数）× vec长度，小于 2410000×10000。而用户的规模是42432500×7712293×300，超出了组件的支持范围，导致内存申请失败。
目前用户需要缩小数据的规模才能计算，并且输入的数据需要分词。

模型数据常见问题

目录

如果以上内容无法解决您的问题，请首先查看**机器学习知识库**，若问题仍得不到解决请粘贴logview（Tensorflow日志中的蓝色链接）到机器学习**工单系统**进行提问。

为什么实验生成的模型为空

现象描述

右键单击模型，选择**查看模型**，结果为空，如下图所示。

解决方法

在机器学习界面单击**设置**，勾选**自动生成PMML**，如下图所示，再次运行即可查看到模型。

如何下载实验生成的模型

在模型菜单中右键单击模型，选择**导出PMML**（PMML是业内标准的模型描述文件，可以通过开源工具解析并使用），如下图所示。

如何在机器学习平台上传数据

数据上传视频：如何上传数据

数据上传文档：数据准备

在线预测功能常见问题

目录

如果以上内容无法解决您的问题，请首先查看机器学习知识库，若问题仍得不到解决请粘贴logview（Tensorflow日志中的蓝色链接）到机器学习工单系统进行提问。

机器学习在线预测说明在哪里

在线预测相关的问题可以参考以下两篇文章：

在线预测功能介绍

模型在线预测

机器学习在线预测只是针对模型的在线预测处理，并不是针对全部流程的在线预测。

AuthorizationFailed错误

子账号调用造成的报错，在线预测调用目前只支持主账号。

kInvalidArgument错误

用户body字段输入错误，请仔细查看模型在线预测文档。

CanNotVisitTheRouter错误

在线预测请求URL错误，请仔细查看模型在线预测文档。

TensorFlow常见问题

目录

如何写入数据到OSS

Tensorflow案例有哪些

其它问题

如果以上内容无法解决您的问题，请首先查看**机器学习知识库**，若问题仍得不到解决请粘贴logview（Tensorflow日志中的蓝色链接）到机器学习**工单系统**进行提问。

如何开通深度学习功能

目前机器学习平台深度学习相关功能处于公测阶段，深度学习组件包含TensorFlow、Caffe、MXNet三个框架。开通方式如下图，进入机器学习控制台，在相应项目下开启GPU资源即可。

开通GPU资源的项目会被分配到公共的资源池，可以动态地调用底层的GPU计算资源。另外需要设置OSS的访问权限，如下图所示。

如何支持多python文件脚本引用

可以通过python模块文件组织训练脚本。将模型定义在不同的Python文件里，将数据的预处理逻辑放在另外

一个Python文件中，最后有一个Python文件将整个训练过程串联起来。
例如在test1.py中定义了一些函数，需要在test2.py文件使用test1.py中的函数，并且将test2.py作为程序入口文件，只需要将test1.py和test2.py打包成tar.gz文件上传即可，如下图所示。

- **Python代码文件**为定义的tar.gz包
- **Python主文件**为定义的入口程序文件

如何上传数据到OSS

详细步骤可参考**如何上传数据**视频。

使用深度学习算法处理数据时，数据先存储到OSS的bucket中。首先要创建OSS Bucket，由于深度学习的GPU集群在**华东2**，建议您创建OSS Bucket时选择**华东2**地区。这样在数据传输时就可以使用阿里云经典网络，算法运行时不需要收取流量费用。Bucket创建好之后，可以在OSS管理控制台创建文件夹、组织数据目录、上传数据。

> OSS支持多种方式上传数据，API或SDK请参见
> https://help.aliyun.com/document_detail/31848.html?spm=5176.doc31848.6.580.a6es2a 。
>
> OSS提供了大量工具来帮助用户更加高效地使用OSS，工具列表请参见
> https://help.aliyun.com/document_detail/44075.html?spm=5176.doc32184.6.1012.XIMMUx
> 。

建议您使用 ossutil 或 osscmd命令行工具，通过命令的方式来上传下载文件，同时支持断点续传。

注意：在使用工具时需要配置 AccessID 和 AccessKey，请登录阿里云管理控制台，并在 Access Key 管理界

面创建或查看。

如何读取OSS数据

Python不支持读取oss数据，因此所有调用python的 Open()、 os.path.exist() 等文件和文件夹操作的函数的代码都无法执行。如Scipy.misc.imread()、numpy.load()等。

通常采用以下两种办法在机器学习平台读取数据。

- 使用**tf.gfile**下的函数，适用于简单地读取一张图片，或者一个文本等，成员函数如下。

```
tf.gfile.Copy(oldpath, newpath, overwrite=False) # 拷贝文件
tf.gfile.DeleteRecursively(dirname) # 递归删除目录下所有文件
tf.gfile.Exists(filename) # 文件是否存在
tf.gfile.FastGFile(name, mode='r') # 无阻塞读取文件
tf.gfile.GFile(name, mode='r') # 读取文件
tf.gfile.Glob(filename) # 列出文件夹下所有文件, 支持pattern
tf.gfile.IsDirectory(dirname) # 返回dirname是否为一个目录
tf.gfile.ListDirectory(dirname) # 列出dirname下所有文件
tf.gfile.MakeDirs(dirname) # 在dirname下创建一个文件夹, 如果父目录不存在, 会自动创建父目录. 如果
文件夹已经存在, 且文件夹可写, 会返回成功
tf.gfile.MkDir(dirname) # 在dirname处创建一个文件夹
tf.gfile.Remove(filename) # 删除filename
tf.gfile.Rename(oldname, newname, overwrite=False) # 重命名
tf.gfile.Stat(dirname) # 返回目录的统计数据
tf.gfile.Walk(top, inOrder=True) # 返回目录的文件树
```

具体请参考tf.gfile模块

- 使用tf.gfile.Glob、tf.gfile.FastGFile、 tf.WhoFileReader() 、tf.train.shuffer_batch()，适用于批量读取文件（读取文件之前需要获取文件列表，如果是批量读取，还需要创建batch）。

使用机器学习搭建深度学习实验时，通常需要在界面右侧设置读取目录、代码文件等参数。这些参数通过“—XXX”（XXX代表字符串）的形式传入，**tf.flags**提供了这个功能。

```
import tensorflow as tf
FLAGS = tf.flags.FLAGS
tf.flags.DEFINE_string('buckets', 'oss://{OSS Bucket}/', '训练图片所在文件夹')
tf.flags.DEFINE_string('batch_size', '15', 'batch大小')
files = tf.gfile.Glob(os.path.join(FLAGS.buckets,'*.jpg')) # 如我想列出buckets下所有jpg文件路径
```

小规模读取文件时建议使用tf.gfile.FastGfile()。

```
for path in files:
file_content = tf.gfile.FastGFile(path, 'rb').read() # 一定记得使用rb读取, 不然很多情况下都会报错
image = tf.image.decode_jpeg(file_content, channels=3) # 本教程以JPG图片为例
```

大批量读取文件时建议使用tf.WhoFileReader()。

```
reader = tf.WholeFileReader()  # 实例化一个reader
fileQueue = tf.train.string_input_producer(files) # 创建一个供reader读取的队列
file_name, file_content = reader.read(fileQueue) # 使reader从队列中读取一个文件
image_content = tf.image.decode_jpeg(file_content, channels=3) # 讲读取结果解码为图片
label = XXX # 这里省略处理label的过程
batch = tf.train.shuffle_batch([label, image_content], batch_size=FLAGS.batch_size, num_threads=4,
capacity=1000 + 3 * FLAGS.batch_size, min_after_dequeue=1000)

sess = tf.Session() # 创建Session
tf.train.start_queue_runners(sess=sess) # 重要!!! 这个函数是启动队列, 不加这句线程会一直阻塞
labels, images = sess.run(batch) # 获取结果
```

部分代码解释如下：

- tf.train.string_input_producer：把**files**转换成一个队列，并且需要 **tf.train.start_queue_runners** 来启动队列。
- tf.train.shuffle_batch参数解释如下：
  - **batch_size**：批处理大小。即每次运行这个batch，返回的数据个数。
  - **num_threads**：运行线程数，一般设置为4。
  - **capacity**：随机取文件范围。比如数据集有10000个数据，需要从5000个数据中随机抽取，那么**capacity**就设置成5000.
  - **min_after_dequeue**：维持队列的最小长度，不能大于**capacity**。

如何写入数据到OSS

- 使用tf.gfile.FastGFile()写入

```
tf.gfile.FastGFile(FLAGS.checkpointDir + 'example.txt', 'wb').write('hello world')
```

- 通过tf.gfile.Copy()拷贝

```
tf.gfile.Copy('./example.txt', FLAGS.checkpointDir + 'example.txt')
```

通过以上两种方法，将数据写入OSS中，生成的文件存储在“输出目录/model/example.txt”下。

Tensorflow案例有哪些

如何使用TensorFlow实现图像分类

- 视频地址：https://help.aliyun.com/video_detail/54948.html
- 文档介绍：https://yq.aliyun.com/articles/72841
- 代码下载：https://help.aliyun.com/document_detail/51800.html

如何使用TensorFlow自动写歌

- 文档介绍：https://yq.aliyun.com/articles/134287
- 代码下载：https://help.aliyun.com/document_detail/57011.html

其它问题

如何查看Tensorflow的相关日志？
具体请参考https://yq.aliyun.com/articles/72841 。

model_average_iter_interval参数在设置两个GPU的时候起到什么作用？

- 如果没有设置**model_average_iter_interval**参数，GPU会运行标准的parallel-sgd，每个迭代都会交换梯度更新。
- 如果**model_average_iter_interval**大于1，就是使用 model Average 方法，间隔若干轮（**model_average_iter_interval**设置数值轮数）两个平均模型参数。

两卡带来的增益是训练速度的提升。