环境：

·系统版本：Win10专业版1809

·是否为移动设备：是，笔记本电脑

·Tensorflow版本：tensorflow-gpu2.0.0-alpha0

·Python版本：Python 3.6.8-Anaconda custom(64-bit)

·CUDA版本：CUDA-V10.0.130

·cuDNN版本：cuDNN-7.5.0

·GPU：Nvidia GTX1050Ti

Bug描述：

在tensorflow里使用keras的Callbacks中的Tensorboard的时候，出现错误：

Epoch 1/50

32/60000 [..............................] - ETA: 11:31 - loss: 2.3852 - acc: 0.1562

---------------------------------------------------------------------------

NotFoundError Traceback (most recent call last)

<ipython-input-4-aadf56b04ffa> in <module>

----> 1 model.fit(x\_train, y\_train, epochs=50, callbacks=[tensorboard\_callback])

2

3 model.evaluate(x\_test, y\_test)

~\Anaconda3\envs\lab\lib\site-packages\tensorflow\python\keras\engine\training.py in fit(self, x, y, batch\_size, epochs, verbose, callbacks, validation\_split, validation\_data, shuffle, class\_weight, sample\_weight, initial\_epoch, steps\_per\_epoch, validation\_steps, validation\_freq, max\_queue\_size, workers, use\_multiprocessing, \*\*kwargs)

863 validation\_steps=validation\_steps,

864 validation\_freq=validation\_freq,

--> 865 steps\_name='steps\_per\_epoch')

866

867 def evaluate(self,

~\Anaconda3\envs\lab\lib\site-packages\tensorflow\python\keras\engine\training\_arrays.py in model\_iteration(model, inputs, targets, sample\_weights, batch\_size, epochs, verbose, callbacks, val\_inputs, val\_targets, val\_sample\_weights, shuffle, initial\_epoch, steps\_per\_epoch, validation\_steps, validation\_freq, mode, validation\_in\_fit, prepared\_feed\_values\_from\_dataset, steps\_name, \*\*kwargs)

361 # Callbacks batch end.

362 batch\_logs = cbks.make\_logs(model, batch\_logs, batch\_outs, mode)

--> 363 callbacks.\_call\_batch\_hook(mode, 'end', batch\_index, batch\_logs)

364 progbar.on\_batch\_end(batch\_index, batch\_logs)

365

~\Anaconda3\envs\lab\lib\site-packages\tensorflow\python\keras\callbacks.py in \_call\_batch\_hook(self, mode, hook, batch, logs)

225 for callback in self.callbacks:

226 batch\_hook = getattr(callback, hook\_name)

--> 227 batch\_hook(batch, logs)

228 self.\_delta\_ts[hook\_name].append(time.time() - t\_before\_callbacks)

229

~\Anaconda3\envs\lab\lib\site-packages\tensorflow\python\keras\callbacks.py in on\_train\_batch\_end(self, batch, logs)

507 """

508 # For backwards compatibility.

--> 509 self.on\_batch\_end(batch, logs=logs)

510

511 def on\_test\_batch\_begin(self, batch, logs=None):

~\Anaconda3\envs\lab\lib\site-packages\tensorflow\python\keras\callbacks\_v1.py in on\_batch\_end(self, batch, logs)

360 self.\_total\_batches\_seen += 1

361 if self.\_is\_profiling:

--> 362 profiler.save(self.log\_dir, profiler.stop())

363 self.\_is\_profiling = False

364 elif (not self.\_is\_profiling and

~\Anaconda3\envs\lab\lib\site-packages\tensorflow\python\eager\profiler.py in save(logdir, result)

141 logdir, 'plugins', 'profile',

142 datetime.datetime.now().strftime('%Y-%m-%d\_%H-%M-%S'))

--> 143 gfile.MakeDirs(plugin\_dir)

144 maybe\_create\_event\_file(logdir)

145 with gfile.Open(os.path.join(plugin\_dir, 'local.trace'), 'wb') as f:

~\Anaconda3\envs\lab\lib\site-packages\tensorflow\python\lib\io\file\_io.py in recursive\_create\_dir(dirname)

446 errors.OpError: If the operation fails.

447 """

--> 448 recursive\_create\_dir\_v2(dirname)

449

450

~\Anaconda3\envs\lab\lib\site-packages\tensorflow\python\lib\io\file\_io.py in recursive\_create\_dir\_v2(path)

462 """

463 with errors.raise\_exception\_on\_not\_ok\_status() as status:

--> 464 pywrap\_tensorflow.RecursivelyCreateDir(compat.as\_bytes(path), status)

465

466

~\Anaconda3\envs\lab\lib\site-packages\tensorflow\python\framework\errors\_impl.py in \_\_exit\_\_(self, type\_arg, value\_arg, traceback\_arg)

546 None, None,

547 compat.as\_text(c\_api.TF\_Message(self.status.status)),

--> 548 c\_api.TF\_GetCode(self.status.status))

549 # Delete the underlying status object from memory otherwise it stays alive

550 # as there is a reference to status from this from the traceback due to

NotFoundError: Failed to create a directory: logs/fit/20190315-164851\plugins\profile\2019-03-15\_16-48-53; No such file or directory

相关代码：

log\_dir="logs/fit/" + datetime.datetime.now().strftime("%Y%m%d-%H%M%S")

tensorboard\_callback = TensorBoard(log\_dir)

model.compile(optimizer='adam',

loss='sparse\_categorical\_crossentropy',

metrics=['accuracy'])

问题解决：

Github链接：<https://github.com/tensorflow/tensorboard/issues/2023>

【

You’re hitting tensorflow/tensorflow#26021,

a Windows-specific bug in TensorFlow.

The fix is to use the platform-appropriate path separators in log\_dir

rather than hard-coding forward slashes:

log\_dir = os.path.join(

"logs",

"fit",

datetime.datetime.now().strftime("%Y%m%d-%H%M%S"),

)

】

tensorflow/tensorflow#26021问题链接：

<https://github.com/tensorflow/tensorflow/issues/26021>

【

Currently, on Unix, SplitPath splits only on forward slashes; on

Windows, it splits on forward slashes unless there are no forward

slashes in the string, in which case it splits on backslashes. This is

confusing, and inconsistent with platform APIs like \_wmkdir, which

interpret both \ and / as valid path delimiters.

For instance, CreateRecursiveDirectory(r"foo\bar/baz\quux") fails on

Windows. This should be broken into ["foo", "bar", "baz", "quux"],

but actually becomes ["foo", "bar", "baz\quux"]. When CreateDir is

called on each of those components in sequence, it creates foo, then

creates foo/bar, then tries to create foo/bar/baz\quux, which fails

because the directory foo/bar/baz does not exist.

(Googlers, see http://b/125762969 for an example of this error.)

The whole situation is a bit less clean than one might hope, because the

TF filesystem API prefers to use /-separators everywhere while people

in Python-land prefer to use os.path.join to insert the proper

platform-specific path separator. But it seems clear that the current

behavior, where the delimiter used depends on what characters are in the

string, is undesirable, and that being consistent with native APIs is a

better choice.

】

修正后代码：

# log\_dir="logs/fit/" + datetime.datetime.now().strftime("%Y%m%d-%H%M%S")

log\_dir = os.path.join(

"logs",

"fit",

datetime.datetime.now().strftime("%Y%m%d-%H%M%S"),

)

tensorboard\_callback = TensorBoard(log\_dir)

model.compile(optimizer='adam',

loss='sparse\_categorical\_crossentropy',

metrics=['accuracy'])

运行结果：

成功运行，并且不报错。