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：

当使用了自定义的优化器的时候，使用tensorflow.keras的model.save()方法的时候，出现错误：

(tf2) D:\Users\suger\Desktop\tf2\hat\_0.1.0>cd d:\Users\suger\Desktop\tf2\hat\_0.1.0 && cmd /C "set "PYTHONIOENCODING=UTF-8" && set "PYTHONUNBUFFERED=1" && D:\Anaconda3\envs\tf2\python.exe c:\Users\suger\.vscode\extensions\ms-python.python-2019.2.5558\pythonFiles\ptvsd\_launcher.py --default --client --host localhost --port 61765 d:\Users\suger\Desktop\tf2\hat\_0.1.0\main.py "

2019-03-26 21:13:56.106495: I tensorflow/core/platform/cpu\_feature\_guard.cc:142] Your CPU supports instructions that this TensorFlow binary was not compiled to use: AVX2

2019-03-26 21:13:56.124981: I tensorflow/stream\_executor/platform/default/dso\_loader.cc:42] Successfully opened dynamic library nvcuda.dll

2019-03-26 21:13:57.051136: I tensorflow/core/common\_runtime/gpu/gpu\_device.cc:1467] Found device 0 with properties:

name: GeForce GTX 1050 Ti major: 6 minor: 1 memoryClockRate(GHz): 1.62

pciBusID: 0000:01:00.0

totalMemory: 4.00GiB freeMemory: 3.30GiB

2019-03-26 21:13:57.070503: I tensorflow/core/common\_runtime/gpu/gpu\_device.cc:1546] Adding visible gpu devices: 0

2019-03-26 21:13:57.802595: I tensorflow/core/common\_runtime/gpu/gpu\_device.cc:1015] Device interconnect StreamExecutor with strength 1 edge matrix:

2019-03-26 21:13:57.818759: I tensorflow/core/common\_runtime/gpu/gpu\_device.cc:1021] 0

2019-03-26 21:13:57.824508: I tensorflow/core/common\_runtime/gpu/gpu\_device.cc:1034] 0: N

2019-03-26 21:13:57.830372: I tensorflow/core/common\_runtime/gpu/gpu\_device.cc:1149] Created TensorFlow device (/job:localhost/replica:0/task:0/device:GPU:0 with 3004 MB memory) -> physical GPU (device: 0, name: GeForce GTX 1050 Ti, pci bus id: 0000:01:00.0, compute capability: 6.1)

=>dat=cifar10 mod=LeNet1

################################

[logs] 2019-03-26-21-13-58

[logs] Datasets: cifar10

[logs] Models: LeNet1

[logs] Epochs: 5

[logs] Batch\_size: 128

[logs] Model Optimizer exist

[logs] h5 not exist, create one

[logs] logs dir: logs\cifar10\_LeNet1\_0\

Train on 50000 samples, validate on 10000 samples

Epoch 1/5

2019-03-26 21:14:23.032117: I tensorflow/core/profiler/lib/profiler\_session.cc:164] Profile Session started.

2019-03-26 21:14:23.048012: I tensorflow/stream\_executor/platform/default/dso\_loader.cc:42] Successfully opened dynamic library cupti64\_100.dll

128/50000 [..............................] - ETA: 11:47 - loss: 2.5079 - accuracy: 0.0547WARNING: Logging before flag parsing goes to stderr.

W0326 21:14:23.359762 6056 callbacks.py:236] Method (on\_train\_batch\_end) is slow compared to the batch update (0.153339). Check your callbacks.

50000/50000 [==============================] - 11s 225us/sample - loss: 2.2438 - accuracy: 0.1572 - val\_loss: 2.0970 - val\_accuracy: 0.2961

Epoch 2/5

50000/50000 [==============================] - 9s 179us/sample - loss: 2.0929 - accuracy: 0.2248 - val\_loss: 1.9330 - val\_accuracy: 0.3577

Epoch 3/5

50000/50000 [==============================] - 9s 178us/sample - loss: 1.9845 - accuracy: 0.2677 - val\_loss: 1.8130 - val\_accuracy: 0.3799

Epoch 4/5

50000/50000 [==============================] - 9s 179us/sample - loss: 1.9116 - accuracy: 0.2938 - val\_loss: 1.7190 - val\_accuracy: 0.4006

Epoch 5/5

50000/50000 [==============================] - 9s 178us/sample - loss: 1.8597 - accuracy: 0.3132 - val\_loss: 1.6864 - val\_accuracy: 0.4143

10000/10000 [==============================] - 2s 153us/sample - loss: 1.6864 - accuracy: 0.4142

[logs] total loss: 1.6864, accuracy: 0.4142

Traceback (most recent call last):

File "c:\Users\suger\.vscode\extensions\ms-python.python-2019.2.5558\pythonFiles\ptvsd\_launcher.py", line 45, in <module>

main(ptvsdArgs)

File "c:\Users\suger\.vscode\extensions\ms-python.python-2019.2.5558\pythonFiles\lib\python\ptvsd\\_\_main\_\_.py", line 357, in main

run()

File "c:\Users\suger\.vscode\extensions\ms-python.python-2019.2.5558\pythonFiles\lib\python\ptvsd\\_\_main\_\_.py", line 257, in run\_file

runpy.run\_path(target, run\_name='\_\_main\_\_')

File "D:\Anaconda3\envs\tf2\lib\runpy.py", line 263, in run\_path

pkg\_name=pkg\_name, script\_name=fname)

File "D:\Anaconda3\envs\tf2\lib\runpy.py", line 96, in \_run\_module\_code

mod\_name, mod\_spec, pkg\_name, script\_name)

File "D:\Anaconda3\envs\tf2\lib\runpy.py", line 85, in \_run\_code

exec(code, run\_globals)

File "d:\Users\suger\Desktop\tf2\hat\_0.1.0\main.py", line 23, in <module>

main()

File "d:\Users\suger\Desktop\tf2\hat\_0.1.0\main.py", line 19, in main

save(model)

File "d:\Users\suger\Desktop\tf2\hat\_0.1.0\ops.py", line 32, in save

model.save(SAVE\_NAME)#, include\_optimizer=not OPT\_EXIST

File "D:\Anaconda3\envs\tf2\lib\site-packages\tensorflow\python\keras\engine\network.py", line 1314, in save

save\_model(self, filepath, overwrite, include\_optimizer)

File "D:\Anaconda3\envs\tf2\lib\site-packages\tensorflow\python\keras\saving\hdf5\_format.py", line 137, in save\_model

save\_optimizer\_weights\_to\_hdf5\_group(f, model.optimizer)

File "D:\Anaconda3\envs\tf2\lib\site-packages\tensorflow\python\keras\saving\hdf5\_format.py", line 650, in save\_optimizer\_weights\_to\_hdf5\_group

name, val.shape, dtype=val.dtype)

File "D:\Anaconda3\envs\tf2\lib\site-packages\h5py\\_hl\group.py", line 139, in create\_dataset

self[name] = dset

File "D:\Anaconda3\envs\tf2\lib\site-packages\h5py\\_hl\group.py", line 371, in \_\_setitem\_\_

h5o.link(obj.id, self.id, name, lcpl=lcpl, lapl=self.\_lapl)

File "h5py\\_objects.pyx", line 54, in h5py.\_objects.with\_phil.wrapper

File "h5py\\_objects.pyx", line 55, in h5py.\_objects.with\_phil.wrapper

File "h5py\h5o.pyx", line 202, in h5py.h5o.link

RuntimeError: Unable to create link (name already exists)

Bug相关代码及操作：

Python main.py

=>dat=cifar10 mod=LeNet1【为了复现bug，写了一个包含自定义优化器的LeNet1】

LeNet1.py

from tensorflow.python.keras.optimizers import \*

LENET1\_OPT = SGD(lr=1e-3, momentum=0.9, decay=0.1/350, nesterov=True)

问题解决：

模型保存的时候，不保存优化器参数。

model.save(SAVE\_NAME, include\_optimizer=not OPT\_EXIST)

注：OPT\_EXIST是args内一个全局布尔变量，动态获取模型是否自定义了优化器，如果有，值为真，并且将该自定义优化器赋值给优化器变量。这里not OPT\_EXIST是取其否定，即当存在自定义优化器，不保存优化器参数。

Bug原因：

Keras的模型保存不会保存自定义优化器的参数，无法写入。

运行结果：

使用自定义优化器的模型可以正常保存，并且可以被正常读取。