Instructions to install TensorFlow in a Conda Environment #153

Open **mwidjaja1** opened this issue on Feb 2, 2021 · 86 comments

mwidjaja1 commented on Feb 2, 2021 • edited

This is not so much an issue as opposed to a 'How To' if you'd like to install this version of Tensorflow in Conda.

Prerequisites: You must be on macOS Big Sur

If you have an Apple Silicon Mac, this is a freebie, you're already on Big Sur. If you're on an Intel Mac, the Intel versions of TensorFlow are Big Sur only.

Sanity Check before Proceeding: To ensure you're on the right version of macOS, run `sw_vers -productVersion` in your terminal. If it's not version 11.##, you're not on Big Sur and must upgrade to it from the macOS App Store.

Prerequisites: Install XCode Command Line Tools

Install Xcode Command Line tools if you haven't. To do so, run this in your terminal: `xcode-select --install`

Sanity Check before Proceeding: To ensure installation worked, run `which xcrun` in your terminal and you should get a path like `/usr/bin/xcrun` . If you haven't, you did not install it correctly.

Prerequisites: Install Miniforge

Where to download Miniforge from
Miniforge, is a 'lightweight' Python interpreter that has full access to the Conda ecosystem. You can download Miniforge from https://github.com/conda-forge/miniforge#miniforge3. You can use Anaconda if you're on Intel, but note that this guide will be written from the perspective of using miniforge.

Sanity Check before Proceeding:

- Run `file $(which python)` in your terminal (thanks to **@lebigot** for this shortcut!). Please make sure that you got:

- If you have an Apple Silicon Mac, it should also say `Mach-O 64-bit executable arm64`. If you have an Intel Mac, it should also say `Mach-O 64-bit executable x86_64`.
- Run `which pip` in your terminal and it too should resolve to some path that implies you're using miniforge3.

If any of those sanity checks failed, you must redo this section. Please ensure that you downloaded the correct Miniforge for your system architecture and installed it. If you did all that, set your environment paths to Miniforge's Python Installation. To do that, you need to figure out where conda was installed to (it's probably `~/miniforge3/condabin/conda`) and then run `~/miniforge3/condabin/conda init` in your terminal.

Apple Silicon Only Warning: You CANNOT use Anaconda

This warning only applies to Apple Silicon Macs. Anaconda comes with many Python packages included, some of which are not Apple Silicon (i.e. ARM) compatible and thus Anaconda is not ARM compatible. You can use Anaconda if you're using an Intel Mac though.

If you were planning to use Anaconda on ARM, please scroll back up and install Miniforge. Miniforge has Conda, which means you can install many of the packages you want such as Pandas, Scipy, and Numpy -- unlike Anaconda, you just have to do the install manually by running `conda install mypackagenamehere`.

Intel Only Warning: Python Bugs in Big Sur

This warning only apply to Intel Macs. For Intel, both Anaconda and MiniForge have a Python Bug which prevents you from running Python correctly in some instances on macOS Big Sur. Until the Python community fixes this, each time prior to loading Python, you must run `export SYSTEM_VERSION_COMPAT=0`. You could also add this to your `.bash_profile` or other shell environment file if you have one, to do this automatically for you.

Installing TensorFlow

Attached to this Issue is a YAML file which will help you create a Conda Environment with TensorFlow, along with all the prerequisites you need from the ARM conda-forge channel.

1. Download environment.yml, which contains the instructions to create a Python environment with the dependencies you need -- we'll install TensorFlow afterwards. Some browsers insist on adding `.txt` to the end of the file -- do not let your browser do that. [thanks to **@isuruf** for streamlining this file to be all Conda]
2. In your terminal run this command, replacing the uppercase variables with the path to your environment.yml file and your desired name for this environment: `conda env create --file=PATH_TO_ENVIRONMENT.YML --name=YOUR_ENV_NAME_HERE`.
3. Activate that environment by running this command, replacing the uppercase variable with your environment's name: `conda activate YOUR_ENV_NAME_HERE`

For X86 as of 03/11/2021:
Thanks to **@edwin-yan** for the updated commands

```
pip install --upgrade --force --no-dependencies
https://github.com/apple/tensorflow_macos/releases/download/v0.1alpha3/tensorflow_macos-
0.1a3-cp38-cp38-macosx_11_0_x86_64.whl
https://github.com/apple/tensorflow_macos/releases/download/v0.1alpha3/tensorflow_addons_
0.1a3-cp38-cp38-macosx_11_0_x86_64.whl
```

For Apple Silicon as of 03/11/2021:

```
pip install --upgrade --force --no-dependencies
https://github.com/apple/tensorflow_macos/releases/download/v0.1alpha3/tensorflow_macos-
0.1a3-cp38-cp38-macosx_11_0_arm64.whl
https://github.com/apple/tensorflow_macos/releases/download/v0.1alpha3/tensorflow_addons_
0.1a3-cp38-cp38-macosx_11_0_arm64.whl
```

5. Finally, give it a spin. Run `python` and try importing `tensorflow`.

Example Commands

In this below example, I'm installing & running the ARM version of tensorflow from an environment I've named `test`. The yml file is placed in the same directory I'm running this command from, which is my home directory (i.e. `~`)

```
conda env create --file=environment.yml --name=test
conda activate test
pip install --upgrade --force --no-dependencies
https://github.com/apple/tensorflow_macos/releases/download/v0.1alpha2/tensorflow_addons_
0.1a2-cp38-cp38-macosx_11_0_arm64.whl
https://github.com/apple/tensorflow_macos/releases/download/v0.1alpha2/tensorflow_macos-
0.1a2-cp38-cp38-macosx_11_0_arm64.whl
python
import tensorflow
```

```
  Downloading https://github.com/apple/tensorflow_macos/releases/download/v0.1alpha2/tensorflow_addons_macos-0.1a2-cp38-cp38-
macosx_11_0_arm64.whl (598 kB)
     |████████████████████████████████| 598 kB 3.6 MB/s
Collecting tensorflow-macos==0.1a2
  Downloading https://github.com/apple/tensorflow_macos/releases/download/v0.1alpha2/tensorflow_macos-0.1a2-cp38-cp38-macosx_
11_0_arm64.whl (124.2 MB)
     |████████████████████████████████| 124.2 MB 5.2 MB/s
Installing collected packages: tensorflow-macos, tensorflow-addons-macos
Successfully installed tensorflow-addons-macos-0.1a2 tensorflow-macos-0.1a2
(test) Matthew@MacBook-M1 ~ python
Python 3.8.6 | packaged by conda-forge | (default, Jan 25 2021, 22:55:00)
[Clang 11.0.1 ] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> import tensorflow
>>>
```

Troubleshooting for importing TensorFlow

- Type in `which python` and then `which pip` in your terminal. Both paths should point to a Python that is **inside the environment** you created in Step 2. If it doesn't, you may not have installed Miniforge correctly, ran Step 2 correctly, and/or may not have ran Step 3.
- Run `python --version` and it should be version 3.8. If it isn't, you most likely did not create or activate your environment correctly, as per Steps 2 & 3. Do those again.
- If python is correctly pointed to the right environment but you cannot import tensorflow, consider running step 5 again just to make sure you installed Tensorflow in the appropriate environment.
- If you are using Intel and got a `not a supported wheel on this platform` error, run `export SYSTEM_VERSION_COMPAT=0` in your terminal and try again. If this works, you'll need to do this everytime you use Python until a Python Bug is resolved.
- **Please verify that you did ALL of the Sanity Checks from the previous section and that they resolve appropriately before posting your issue here.** If you do post your issue, please provide the terminal outputs from those steps and bonus points if you share the results of your Sanity Check and run `pip` with a `-v` flag for additional logging. Remember I'm just a volunteer -- I'll try to help but there's only so much I can help with.

Troubleshooting for setting up TensorFlow

- For those having issues with tf.keras.models.load_model about a `failed to decode` error: Try downgrading to h5py to the 2.10.0 wheel file that was packaged with this alpha release (`pip install ~/path to h5py.whl`). Thanks to **@ramicaza**.

👍 68 🎉 23 ❤️ 33 🚀 11


mwidjaja1 mentioned this issue on Feb 2, 2021

Fixing Install Script Typo & adding Conda Install Support #63

Open


mwidjaja1 changed the title ~~Installing TensorFlow in a Conda Environment~~ **Instructions to install TensorFlow in a Conda Environment** on Feb 2, 2021

Closed

thomastiotto commented on Feb 3, 2021

I couldn't install the wheels without renaming them to read "...-macosx_10_16_..." instead of "...-macosx_11_0_..."

mwidjaja1 commented on Feb 3, 2021

@Tioz90 I didn't have that problem. Could you tell me what your Mac's version is? That's Apple (in the Menu Bar) > About my Mac. Mine is Version 11.2.

thomastiotto commented on Feb 3, 2021

@mwidjaja1 MacOs 11.1

mwidjaja1 commented on Feb 3, 2021

@Tioz90 You may want to create a new issue then about this. I'm not on the team working on this, I just wrote up these instructions as a 'side contributor' with their help.

thomastiotto commented on Feb 3, 2021

I think it's a known issue with `pip`

mwidjaja1 commented on Feb 3, 2021

That's an interesting thought @Tioz90 -- My pip version is 20.2.4 which I checked by running `pip --version` . If yours isn't that version, could you get that version and try again?

thomastiotto commented on Feb 3, 2021

Mine is `21.0.1` and it still does not work..

mwidjaja1 commented on Feb 3, 2021

wouldn't know how to begin figuring that out. Glad you got it to work by renaming it though

hoytak commented on Feb 3, 2021

There's a few things that could make this issue a mess. I haven't found versions of pip later than 20.2.4 to work here; version 20.3 and later actually broke some of the logic. Also, the logic inside of pip is duplicated across setuptools and packaging, both of which are baked into the core python distribution; upgrading these might help, but your mileage may vary.

Stanfording commented on Feb 4, 2021 • edited

Does it mean that installing TensorFlow in a Conda Environment allows m1 chip to use tensorflow with SciPy and dependent packages?

thomastiotto commented on Feb 4, 2021

Yes, SciPy works for me

Thomas Tiotto
...

AnnoGooG commented on Feb 4, 2021 • edited

I got this Error
`ERROR: tensorflow_macos-0.1a2-cp38-cp38-macosx_11_0_x86_64.whl is not a supported wheel on this platform.`
This is my System Information:

- python: `/Users/gyuannn/miniforge3/envs/test/bin/python , 3.8.6`
- pip: `/Users/gyuannn/miniforge3/envs/test/bin/pip , 20.2.4`
- System: `MacOS11.2`
- Laptop: MBP16,2019
- Run command on Conda Env builded from `environment.yml`

I'm Confused...

Run `python -m pip debug --verbose` and
find pip not support `cp38-cp38-macosx_11_0_x86_64` only support `cp38-cp38-macosx_10_16_x86_64`

Indeed:
Python 3.8 from the system:

Python 3.9 installed with homebrew:

```
/usr/local/bin/python3 -c "import platform; print(platform.platform())"
macOS-11.0.1-x86_64-i386-64bit
```

Python 3.8 from a conda-forge env:

```
/Users/ogrisel/miniforge3/envs/py38/bin/python -c "import platform;
print(platform.platform())"
macOS-10.16-x86_64-i386-64bit
```

So conda-forge's Python is indeed pretending to be installed on a fake 10.16 version of macOS to bypass this problem with pip 20.2.4.

mwidjaja1 commented on Feb 4, 2021 • edited

@Tioz90, I think @gyuannnn figured it out. I get your exact error on Anaconda on Intel too. Interestingly enough on an identical macOS version on an Apple Silicon Mac, it actually is 11_0. So it looks like it's all Conda Pythons (since I tried on Anaconda and you tried Miniforge) on Big Sur Intel, the platform is being saved as 10_16 rather than 11_0.

Not sure how to report this to the Miniforge folks but I can poke around unless you have an idea where the problem might be. In the meanwhile, can you try downloading the wheel files and renaming it so instead of having `11_0` in the two wheel file names, it says `10_16` instead and try pip installing those? If that works, I'll update the Readme for this workaround.

@hoytak tagging you for situational awareness that we figured out the issue.

EDIT: Nevermind I see you reported it already. https://github.com/conda-forge/miniforge/issues/105. Not sure if this is the right place admittedly (it could be Conda's fault) but let's see if they know where to direct us to.

AnnoGooG commented on Feb 4, 2021

> @Tioz90, I think @gyuannnn figured it out. I get your exact error on Anaconda on Intel too. Interestingly enough on an identical macOS version on an Apple Silicon Mac, it actually is 11_0. So it looks like it's all Conda Pythons (since I tried on Anaconda and you tried Miniforge) on Big Sur Intel, the platform is being saved as 10_16 rather than 11_0.
>
> Not sure how to report this to the Miniforge folks but I can poke around unless you have an idea where the problem might be. In the meanwhile, can you try downloading the wheel files and renaming it so instead of having `11_0` in the two wheel file names, it says `10_16` instead and try pip installing those? If that works, I'll update the Readme for this workaround.

> right place admittedly (it could be Conda's fault) but let's see if they know where to direct us to.

Downloaded and renamed it to 10_16_ works fine.

AnnoGooG commented on Feb 5, 2021

@mwidjaja1
according to conda-forge/miniforge#105
`export SYSTEM_VERSION_COMPAT=0` before running python (on env) can set it to 11_0_

👍 2

thomastiotto commented on Feb 5, 2021

> @Tioz90, I think @gyuannnn figured it out. I get your exact error on Anaconda on Intel too. Interestingly enough on an identical macOS version on an Apple Silicon Mac, it actually is 11_0. So it looks like it's all Conda Pythons (since I tried on Anaconda and you tried Miniforge) on Big Sur Intel, the platform is being saved as 10_16 rather than 11_0.
>
> Not sure how to report this to the Miniforge folks but I can poke around unless you have an idea where the problem might be. In the meanwhile, can you try downloading the wheel files and renaming it so instead of having `11_0` in the two wheel file names, it says `10_16` instead and try pip installing those? If that works, I'll update the Readme for this workaround.
>
> **@hoytak** tagging you for situational awareness that we figured out the issue.
>
> EDIT: Nevermind I see you reported it already. conda-forge/miniforge#105. Not sure if this is the right place admittedly (it could be Conda's fault) but let's see if they know where to direct us to.

Yes, renaming the files installs them without an issue

mwidjaja1 commented on Feb 5, 2021

@gyuannnn Thanks for the heads up, I updated the documentation to reflect this temporary workaround. Python 3.8.8 sounds like it'll fix this issue fortunately.

Conversely, renaming the files may crash your Python @Tioz90 if you ever update your Python to 3.8.8 which is planned to fix this issue. https://www.python.org/downloads/release/python-387/

@hoytak A heads up for your team's awareness that this is an issue... and actually might explain some of the crazy pip commands I noticed you guys pulling off. I'm starting to wonder if some of the weird installation issues I saw, while trying to use Conda installs to setup TensorFlow's pre-reqs, were related to this whole `10_16` vs. `11_0` mishap.

This was referenced on Feb 5, 2021

Installation fails with numpy error #121

Open

Cant start tensorflow on M1: mach-o, but wrong architecture #146

Open

icenando commented on Feb 6, 2021 • edited

How can I "point" python and pip to use their respective miniforge versions? I have my conda environment working fine, but when I enter `which python` and `which pip` I get `/usr/bin/python` and `/usr/bin/pip` respectively.

I tried installing as above but got the wheel error:
`ERROR: tensorflow_macos-0.1a2-cp38-cp38-macosx_11_0_x86_64.whl is not a supported wheel on this platform.`

mathieuboudreau commented on Feb 6, 2021

> How can I "point" python and pip to use their respective miniforge versions? I have my conda environment working fine, but when I enter `which python` and `which pip` I get `/usr/bin/python` and `/usr/bin/pip` respectively.
>
> I tried installing as above but got the wheel error:
> `ERROR: tensorflow_macos-0.1a2-cp38-cp38-macosx_11_0_x86_64.whl is not a supported wheel on this platform.`

+1 Ran into the exact same issue today.

mwidjaja1 commented on Feb 6, 2021 • edited

@icenando & **@mathieuboudreau** this means Miniforge is not added to your system path. This is usually a Googling task just because everybody's terminals shell prompts are different, it'd be way beyond the scope of this post to cover every permutation.

In theory running `conda init` in your terminal should fix it. If it complains it cannot find `conda`, that means you need to figure out where you installed it to and run this command again from that directory where `conda` resides. If that doesn't work you'll need to identify where the Python executable that Miniforge installed is, than figure out what Shell Language your terminal is using by running `echo $SHELL`, and then Googling how you'd add the directory containing Python to your system paths environment variable.

@**mwidjaja1** Thank you for that. `conda init` sorted the python and pip issues, but I'm still getting `ERROR: grpcio-1.33.2-cp38-cp38-macosx_10_16_arm64.whl is not a supported wheel on this platform.` (I tried renaming the file to 10_16 like others recommened).

I notice that the python version installed with miniforge is 3.9. So I tried changing the pip install line to python3.9 instead of 3.8, but I got the same wheel error.

When I try using Apple's .sh file, `ERROR: Python version in specificed virtual environment /Users/nandom/miniforge3/envs/TF not 3.8. Python 3.8 required for tensorflow_macos 0.1a2. .`

Any help to sort this out would be appreciated.

icenando commented on Feb 7, 2021 • edited

Ok, so removing the evironment and re-creating it, specifying `python=3.8` solved it. There were other steps that I had to adapt. This post is the only thing that actually worked (I had to change the filenames in the pip installs, as they have changed since it was written): https://claytonpilat.medium.com/tutorial-tensorflow-on-an-m1-mac-using-jupyter-notebooks-and-miniforge-dbb0ef67bf90

mwidjaja1 commented on Feb 7, 2021

@**icenando** I'm glad you got it working though just to get feedback, I'm not sure why that article would work whereas these notes wouldn't. "Installing TensorFlow" section of this guide tells you to download environment.yml and then activate said environment. That environment is designed to give you Python 3.8 and all of the prerequisites you need.

With all due respect, your first comment implies you didn't read my instructions, because you're right miniforge installs Python 3.9 by default but my instructions literally guide you away from using it. But if you have any error messages while setting up that environment or activating it, please let me know, I'd be glad to rephrase the guide. Thanks.

icenando commented on Feb 7, 2021

@**mwidjaja1** Sorry, I didn't mean to imply that your instructions are incorrect. I'm sure I'm making a lot of mistakes as I go: I'm not that experienced, and there are a lot of instructions that I follow blindly because I don't understand what they do.

I don't remember getting any error messages prior to the wheel one. But regardless, that meant that I was unable to finish the install. Not that I know what I'm talking about, but wouldn't it be good to replace `conda env create --file=PATH_TO_ENVIRONMENT.YML --name=YOUR_ENV_NAME_HERE` with `conda env create --file=PATH_TO_ENVIRONMENT.YML --name=YOUR_ENV_NAME_HERE python=3.8` , just to be sure that the correct version is in use in the newly created environment?

Apologies again if my previous message implied that there was something wrong with your instructions.

mwidjaja1 commented on Feb 7, 2021

@icenando No worries! I've been there too -- I just have been getting more DMs about this and I'm just a 'volunteer' so I've been running around heh.

If you wouldn't mind, could you give the original command another whirl? So long as you name that Conda environment something different than what you did with your working tensorflow env, they shouldn't stomp each other out. Specifically the commands I'm asking you to run are:

```
conda env create --file=PATH_TO_ENVIRONMENT.YML --name=YOUR_ENV_NAME_HERE
conda activate YOUR_ENV_NAME_HERE
python --version
```

The YML file you're downloading is the 'recipe' that Conda will use to download everything and set Python up. Because of that, there SHOULD be no need to specify `python=3.8` in that command because it's in the environment file. BUT, I could definitely be wrong so if that doesn't work, I'd be glad to look at the error some more.

61 hidden items

Load more...

kmdalton commented on Apr 15, 2021

@Eunchan24, the base Tensorflow 2 distribution currently supports Python 3.6-3.8.

manjushribuddha commented on Apr 15, 2021

I find the instructions to install tensorflow on the Mac absolutely useless
...

👎 2

RomainBsb commented on Apr 20, 2021

> not the package executable themselves. I'd be surprised if that path actually exists (and if it does, I'd be surprised how it got there in the first place). Something tells me that in your .bashrc/.bash_profile/.zshrc file, you have an export PATH that shouldn't be there, that's tricking pip to look in the wrong location. **@FedericoMz** posted a comment that would probably help you, though to be honest, based on what you shared, I feel like that's just putting more duct tape on an environment file that's not quite correctly configured.

@mwidjaja1 thank you so much for your answer and sorry for the late reply. It's strange because when I do: `which pip` I get `/Users/romainbesombes/miniforge3/envs/tf_env/bin/pip`

```
(tf_env) ➜ ~ echo $PATH
/Users/romainbesombes/miniforge3/envs/tf_env/bin:/Users/romainbesombes/miniforge3/bin:/Users/romainbesombes/miniforge3/condabin:/usr/local/bin:/usr/bin:/bin:/usr/sbin:/sbin
```

This is what is inside of **.bash_profile** :

```
# >>> conda initialize >>>
# !! Contents within this block are managed by 'conda init' !!
__conda_setup="$('/Users/romainbesombes/miniforge3/bin/conda' 'shell.bash' 'hook' 2> /dev/null)"
if [ $? -eq 0 ]; then
eval "$__conda_setup"
else
if [ -f "/Users/romainbesombes/miniforge3/etc/profile.d/conda.sh" ]; then
. "/Users/romainbesombes/miniforge3/etc/profile.d/conda.sh"
else
export PATH="/Users/romainbesombes/miniforge3/bin:$PATH"
fi
fi
unset __conda_setup
# <<< conda initialize <<<
```

This is what is inside of **.zshrc** :

```
# If you come from bash you might have to change your $PATH.
# export PATH=$HOME/bin:/usr/local/bin:$PATH

# Path to your oh-my-zsh installation.
export ZSH="/Users/romainbesombes/.oh-my-zsh"

source ~/.bash_profile
[...]
```

There is nothing in **.bashrc**

I don't see anything strange except 'source ~/.bash_profile' that is something that I added myself in **.zshrc** to make oh-my-zsh work.
Thanks again for your help

@**RomainBsb** without access to your Mac, it'd be hard for me to guess every possible outcome. This sounds like an issue with something in how you configured your computer and there's only so much I can help you with then. Other things to jot your mind:

- Try creating a brand new empty conda environment, install Python + Pip there, and then try to pip install something. If that works, conda install everything the .yaml file would normally have you install, and then try pip installing TensorFlow again.
- Are you sure you don't have `pip` linked to that random `site-packages` path in the one of the Anaconda bin folders you had in `$PATH` ?
- Try running `grep -ri pip ~` in your terminal -- I'm curious if you have a file that has the word 'pip' in it, linked to something else.
- The most extreme solution, create a brand new user account on your Mac and try installing Miniforge there. If you're able to get pip working there, it is almost certainly a problem with your original account's Python configuration. I wouldn't know what the issue is, but at least then, you can compare files across both user accounts and start to triage where things might have went wrong.

Sorry for the unsatisfying response but again, there's nothing you're showing me that's outright wrong. There's just something, somewhere, incorrectly directing your pip to site-packages rather than the bin folder. Best of luck, sorry I can't help more from a far.

👍 1

 **arge-7** mentioned this issue on Apr 23, 2021

Cannot open jupyter notebook or import tensorflow on Mac M1 #242

Open

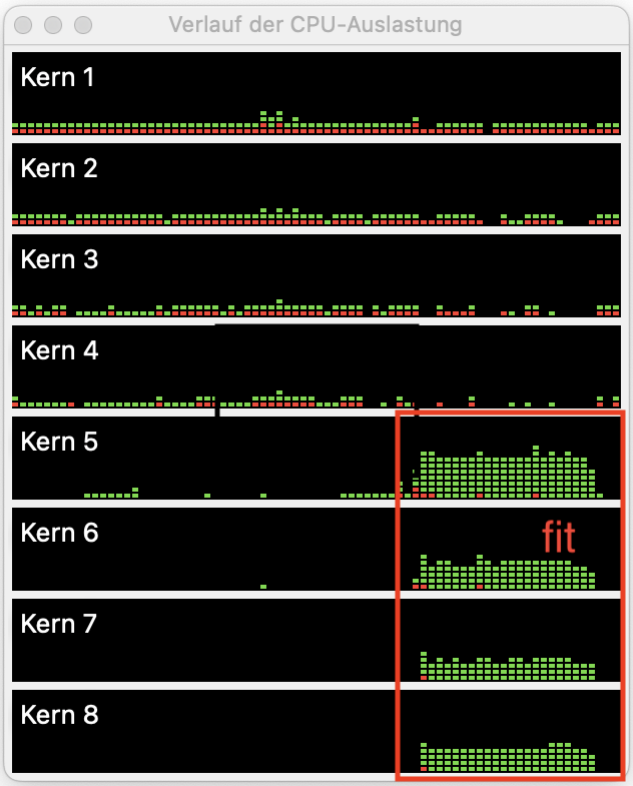
ManuelSchneid3r commented on Apr 27, 2021 • edited

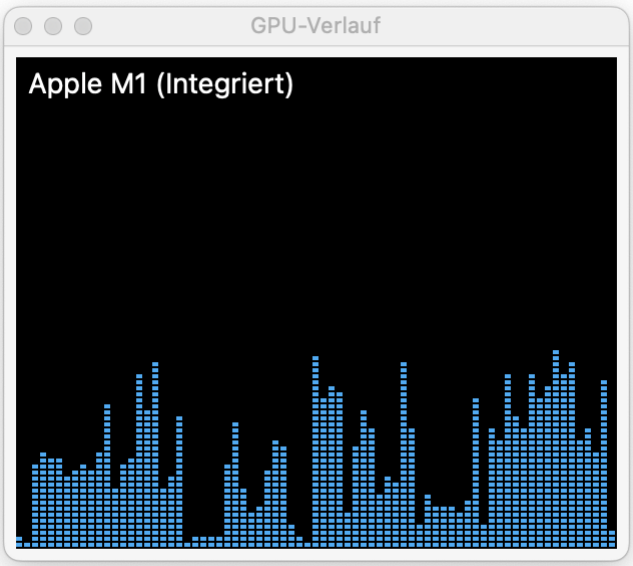
Hi @**mwidjaja1**, I set up my env as you described. I encountered some nonblocking problems.

```
(gpu-env) ~ pip check
tensorflow-macos 0.1a3 has requirement gast==0.3.3, but you have gast 0.4.0.
tensorflow-macos 0.1a3 has requirement grpcio~=1.32.0, but you have grpcio 1.37.0.
tensorflow-macos 0.1a3 has requirement h5py~=2.10.0, but you have h5py 3.2.1.
tensorflow-macos 0.1a3 has requirement numpy~=1.19.2, but you have numpy 1.20.2.
tensorflow-macos 0.1a3 has requirement protobuf~=3.13.0, but you have protobuf 3.15.8.
tensorflow-macos 0.1a3 has requirement tensorflow-estimator~=2.3.0, but you have
tensorflow-estimator 2.4.0.
```

Ain't this a problem?

Futhrer my setup seems to ignore the GPU.





How can I make tensorflow use the GPU instead of the CPU?

This is the cifar10 script to test tensor flow which also reveals that tensorflow does not see the GPU at all.

```
import tensorflow as tf
print(tf.__version__)
```

```
print([x.name for x in device_lib.list_local_devices() if x.device_type == 'GPU'])
from tensorflow.python.compiler.mlcompute import mlcompute
print("is_apple_mlc_enabled %s" % mlcompute.is_apple_mlc_enabled())
print("is_tf_compiled_with_apple_mlc %s" % mlcompute.is_tf_compiled_with_apple_mlc())
from tensorflow.keras import datasets, layers, models

(train_images, train_labels), (test_images, test_labels) = datasets.cifar10.load_data()
train_images, test_images = train_images / 255.0, test_images / 255.0
class_names = ['airplane', 'automobile', 'bird', 'cat', 'deer',
               'dog', 'frog', 'horse', 'ship', 'truck']
model = models.Sequential()
model.add(layers.Conv2D(32, (3, 3), activation='relu', input_shape=(32, 32, 3)))
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Conv2D(64, (3, 3), activation='relu'))
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Conv2D(64, (3, 3), activation='relu'))
model.add(layers.Flatten())
model.add(layers.Dense(64, activation='relu'))
model.add(layers.Dense(10))
model.summary()
model.compile(optimizer='adam',
              loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True),
              metrics=['accuracy'])
history = model.fit(train_images, train_labels, epochs=10,
                    validation_data=(test_images, test_labels))
```

What am I doing wrong?

mwidjaja1 commented on Apr 27, 2021

@hoytak have the requirements for Apple's Tensorflow changed?

@ManuelSchneid3r for your second issue, please make a new GitLab issue (or search the issues list... I'd be surprised if you're the only one having this problem) for the CPU vs. GPU question. This Gitlab issue is specifically for conda installation.

ManuelSchneid3r commented on Apr 29, 2021

@mwidjaja1 indeed there *are* open issues concerning the GPU. I followed your instructions for conda. Does you setup use the CPU? I mean I should get GPU acceleration if you do right?

 **ManuelSchneid3r** mentioned this issue on Apr 29, 2021

The new Apple M1 MLcompute Tensorflow2.4 not compatible with Numpy 1.20.1, after attempting installing ~>1.19.2, got error #220

CarGod commented on Apr 29, 2021

If you get a mistake: `ERROR: grpcio-1.33.2-cp38-cp38-macosx_10_16_arm64.whl is not a supported wheel on this platform.`

you need to uninstall `conda` and install: https://github.com/conda-forge/miniforge#miniforge3

👍 2 🎉 1 ❤️ 1

johnnynunez commented on May 2, 2021 • edited

Python 3.9 is native with Big Sur and TensorFlow v2.5.0rc2 supports python 3.9...
I have checked the binaries one by one that is ARM-compatible in version 3.9.
If there are problems with pip just downgrade to version 19.0.3
If you can't install, use brew install numpy, pandas...
#250

HiLiHeYi commented on May 5, 2021

Hey, first of all, thanks for the instruction it works for me(pic 1). But once I try to run `pip install requirement.txt` which requires `tensorflow>=2.4.0` , it shows me no tensorflow version is available. Can someone help thank you very much

```
~/desktop/Graph_local/GraphSAGE_online master !3 python 1 err Spiderman py
Python 3.8.8 | packaged by conda-forge | (default, Feb 20 2021, 15:50:57)
[Clang 11.0.1 ] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> import tensorflow as tf

print(>>>
>>> print(tf.__version__)
2.4.0-rc0
>>> exit()
```

```
ERROR: Could not find a version that satisfies the requirement tensorflow>=2.4.0 (from ve
rsions: none)
ERROR: No matching distribution found for tensorflow>=2.4.0
```

👀 1

mwidjaja1 commented on May 6, 2021

@HiLiHeYi I think your issue might be better as its own GitLab Issue. I'm just a 'volunteer' monitoring this issue just for Conda installation issues, and this guide has no reference to a requirement.txt file. It sounds like you're having 'other' issues using TensorFlow for your specific Pythonic use case, as opposed to installing TensorFlow, which I see you got working.

even mention tensorflow because you already had it installed anyway. I think that should do the trick.

If this doesn't do the trick or doesn't meet your needs, please create a new GitLab issue and tag me so we can discuss there, and keep this clear for Conda install issues. Thanks!

ryanrudes mentioned this issue on May 9, 2021

Running both TensorFlow and gym on Apple Silicon openai/gym#2134

Closed

nsameerr mentioned this issue on May 16, 2021

Apple M1 TensorFlow #263

Open

johnnynunez commented on May 17, 2021

tensorflow v2.5.0 is out. Is it compatible with m1?
Support native python 3.9...

JakobLS commented on May 24, 2021 • edited

Failing with installing Miniforge on Mac M1 using brew

> If any of those sanity checks failed, you must redo this section. Please ensure that you downloaded the correct Miniforge for your system architecture and installed it. If you did all that, set your environment paths to Miniforge's Python Installation. To do that, you need to figure out where conda was installed to (it's probably ~/miniforge3/condabin/conda) and then run ~/miniforge3/condabin/conda init in your terminal.

My installation was failing on this part. It turned out that Miniforge had been installed to `/opt/homebrew/Caskroom/miniforge` , so by running `/opt/homebrew/Caskroom/miniforge/base/condabin/conda init` followed by `source .bash_profile` solved the problem for me.

mwidjaja1 commented on May 26, 2021

Hey **@JakobLS**, while I'm glad you figured it out, I'm super curious how your Homebrew got installed to caskroom. I... don't think that was possible. Do you recall manually requiring miniforge to save there or did it just happen?

Glad you figured it out though, that is the absolute right step! Thanks for sharing

JakobLS commented on May 26, 2021

Hi **@mwidjaja1**. Thank you so much for supporting the community. I don't recall the exact details, but I'm pretty sure I didn't manually specify miniforge to be installed at a certain location, nor with Homebrew. I followed your steps in the top, but chose to install miniforge using Homebrew purely for convenience (rather than downloading it first).

And this tutorial was very useful to install Homebrew with.

I had to add `export PATH=$PATH:/opt/homebrew/bin` and `export PATH="/opt/homebrew/Caskroom/miniforge/base:$PATH"` to `.bash_profile` for Homebrew and miniforge respectively though.

Running on MacOS Big Sur 11.3.1

jaismith mentioned this issue on May 30, 2021

op type not registered `NormalizeUTF8` initializing BERT #276

Open

scan mentioned this issue on Jun 6, 2021

Check failed: outputs_[index].tensor == nullptr #280

Open

backyardbiomech mentioned this issue on Jul 5, 2021

`line 3: 76717 Illegal instruction: 4` --> 2.2rc3 will not install on M1 Macs
DeepLabCut/DeepLabCut#1380

Closed

shakewingo mentioned this issue on Jul 9, 2021

Can this package be compatible with MacOS M1 chip? lda-project/lda#115

Closed

tetsuyasu mentioned this issue on Aug 25, 2021

tensorflow-addons doesn't work with tensorflow-macos tensorflow/addons#2503

Closed

andres-zartab mentioned this issue on Sep 16, 2021

Open

ms003 mentioned this issue on Sep 21, 2021

Installation fails for MacOs M1 miniforge spyder-ide/spyder#16463

Closed

mbuttner mentioned this issue on Oct 25, 2021

Cannot import sccoda.util.comp_ana theislab/scCODA#39

Closed

EliasLum mentioned this issue on Nov 15, 2021

Error in asteroid.engine.System() for M1 asteroid-team/asteroid#573

Closed

htrivino20 mentioned this issue on Nov 21, 2021

Conflicting dependencies (Apple Silicon M1 hardware Monterey 12.0.1) ANTsX/ANTsPyNet#43

Open

gcroci2 mentioned this issue on Feb 7

Fix for installation on macbooks with M1 chips dianna-ai/dianna#200

Closed

Assignees

No one assigned

Labels

None yet

Projects

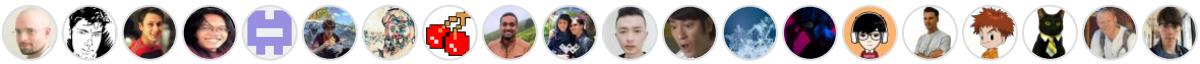
None yet

Milestone

Development

No branches or pull requests

32 participants

 and others