

How to use Pegasus on HCC server

SD2019-Plant Phenotyping

Dependencies

- pegasus (does not need to be installed)
- pandas
- matplotlib
- scipy(version 1.1.0)
- scikit-learn
- pillow
- opencv
- tensorflow
- keras
- numpy

Cloning Project

1. You can clone the project using:

```
git clone git@github.com:cseniordesign/plant-phenotyping.git
```

Transferring Data to HCC Server

2. This workflow can only be run on the HCC server, so move the whole project (plant-phenotyping) into your work directory on the HCC server. Transferring data instructions below:
 - a. Mac/Linux (under File Transferring With HCC Supercomputers):
https://hcc.unl.edu/docs/quickstarts/connecting/for_maclinux_users/
 - b. Window (under File Transferring With HCC Supercomputers):
https://hcc.unl.edu/docs/quickstarts/connecting/for_windows_users/
3. Ensure that the **schnablelab** project (<https://github.com/freemao/schnablelab>) is on in the pics2predictions folder .
4. Ensure that the plant dataset is on the HCC server and is accessible by you.

Creating Anaconda Environment to Run Workflow

5. Install dependencies: [pandas, matplotlib, scipy(version 1.1.0), scikit-learn, pillow, opencv, tensorflow, keras, and numpy]
 - a. Load anaconda:

```
module load anaconda
```

- b. Create your own environment:

```
conda create -n [env_name]
```

(Where [env_name] is the name that you want your environment to be)

- c. To activate the environment:

```
conda activate [env_name]
```

(You can verify you have activated your environment by looking to the left username in the terminal. It should show your environment name in parentheses).

To deactivate:

```
conda deactivate
```

- d. Install dependencies [Pandas, Matplotlib, Scipy(version 1.1.0), scikit-learn, Pillow, opencv, tensorflow, Keras, numpy]:

```
conda install [package_name]
```

(must activate your environment to use this line)

Modifying Workflow Paths to Work with File Structure

6. Modify **rc.txt** located in the pics2predictions folder: Each line in rc.txt is just the [file name] [path to file]. Change the path to the model file (.h5) to the path to match to your work directory's model file. (eg. file:///work/[groupname]/[username]/[path to project (if applicable)]/plant-phenotyping/pics2predictions/input/model_4_300_3.40421104694053e-05.h5 site="local-hcc")
7. Modify **tc.txt** located in the pics2predictions folder: Change pfn under the tr python3 to match the file path to your work directory's run_python.sh script. (eg. file:///work/[groupname]/[username]/[path to project (if applicable)]/plant-phenotyping/pics2predictions/scripts/run_python.sh site="local-hcc")
8. Modify **daxgen.py** located in the pics2predictions folder: Change the definition of path list to the file path to the dataset's Hyp_SV_90 folders. ex:

```
path_list = glob("/work/csesd/pnnguyen/data/*/Hyp_SV_90/")
```

9. Modify **run_python.sh** located in pics2predictions' scripts folder: Change **conda activate numpy** to **conda activate [env_name]** (where [env_name] is the name of the environment you created) and change the python path (content

after export PYTHONPATH=) to the path to the schnablelab project. ex: (Note schnablelab folder is in pics2predictions)

```
export PYTHONPATH=/work/csesd/pnnguyen/pics2predictions:$PYTHONPATH
```

Running Workflow

10. Run the command: `./generate_dax.sh [dax file name].dax`
11. Then the command : `./plan_dax.sh [dax file name].dax`
12. Use **pegasus-status -l ...**(copy this from the output) to see the status. (optional)
 - a. ex: `pegasus-status -l`
`/work/csesd/johnsuzh/pics2predictions/submit/johnsuzh/pegasus/split/run0010`
13. Use **pegasus-remove ...**(copy this from the output) to remove the current running workflow. (optional)
 - a. `pegasus-remove`
`/work/csesd/johnsuzh/pics2predictions/submit/johnsuzh/pegasus/split/run0010`
14. Use **pegasus-analyzer ...**(copy this from the output) for a failing workflow to see the error message
 - a. `pegasus-analyzer`
`/work/csesd/johnsuzh/pics2predictions/submit/johnsuzh/pegasus/split/run0010`
15. When it's done, the status should be 100% in the %DONE column. And you can see the output in the output folder.

```
[johnsuzh@login.crane pics2predictions]$ pegasus-status -l /work/csesd/johnsuzh/pics2predictions/submit/johnsuzh/pegasus/split/run0013
STAT IN_STATE JOB
Run 07:44:37 split-0 ( /lustre/work/csesd/johnsuzh/pics2predictions/submit/johnsuzh/pegasus/split/run0013 )
Summary: 1 Condor job total (R:1)

UNRDY READY PRE IN_Q POST DONE FAIL %DONE STATE DAGNAME
0 0 0 0 0 14,571 0 100.0 Success *split-0.dag
Summary: 1 DAG total (Success:1)
```

```
[[johnsuzh@login.crane output]$ ls -l | wc -l
6622
```

[illegible]

16. When you want to run the workflow again make sure to remove the contents in the output folder

17. Note: This currently only works for corn images. To use hyperspectral images for other plants, add the prediction model into the input folder and change **rc.txt** to have that file and its path.