

Manul

1. Preparation

Step 0: Get some BGM

Step 1: Download training and testing data

<https://www.kaggle.com/datasets/vsdf2898kaggle/osocrtraining>

Step 2: Setting up environment

https://github.com/lancercat/make_env/

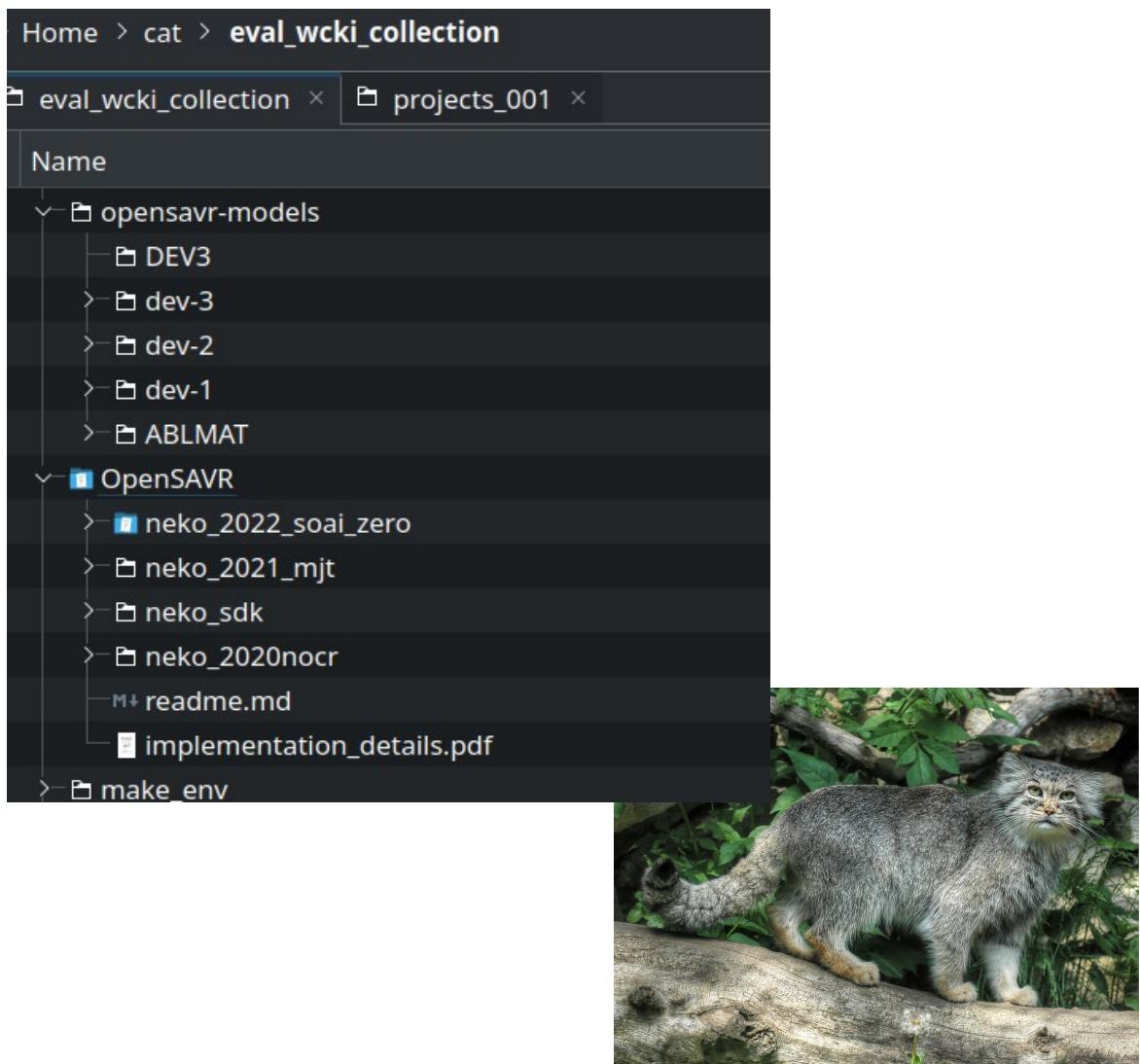
Step 3: Download code

<https://github.com/lancercat/OpenSAVR>

Step 4: Download pretrained models

(Well the link is for now---ummm--private unless I provide a phone number...)

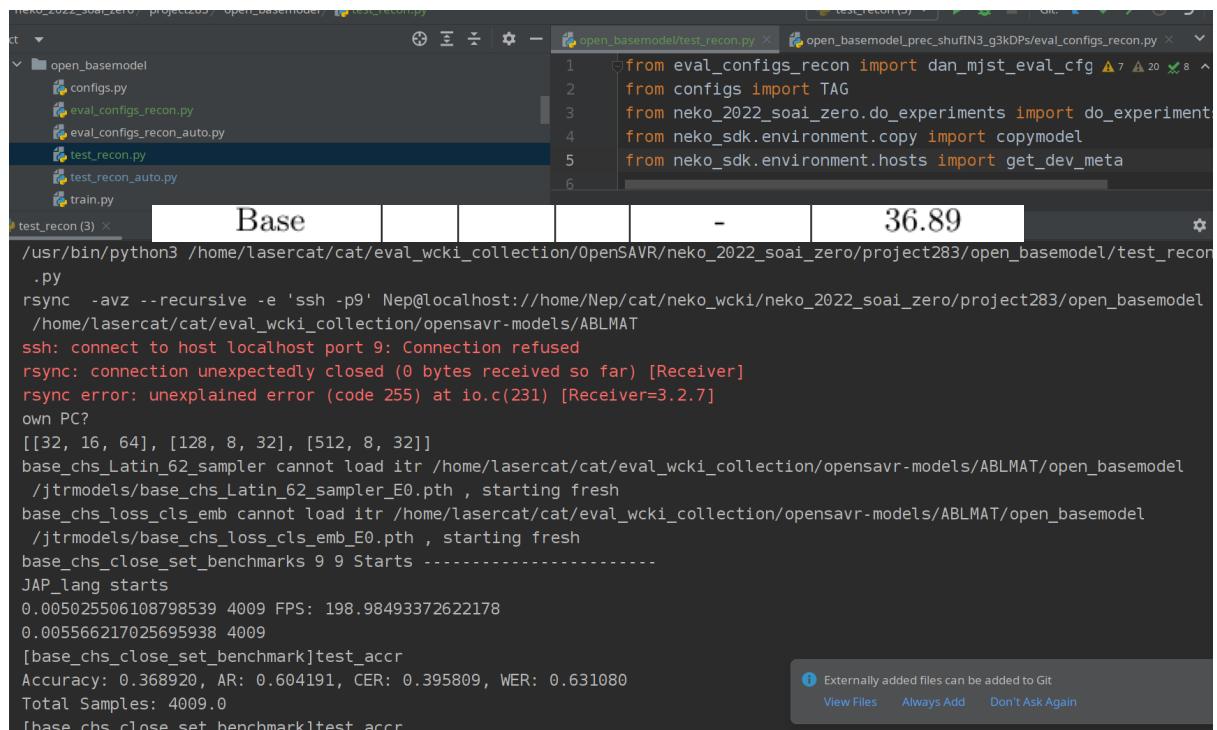
Step 5: Check that you have the files illustrated in the figure below.



2. Running the Ablative & Sensitivity Experiments

Step 1: Run basemodel:

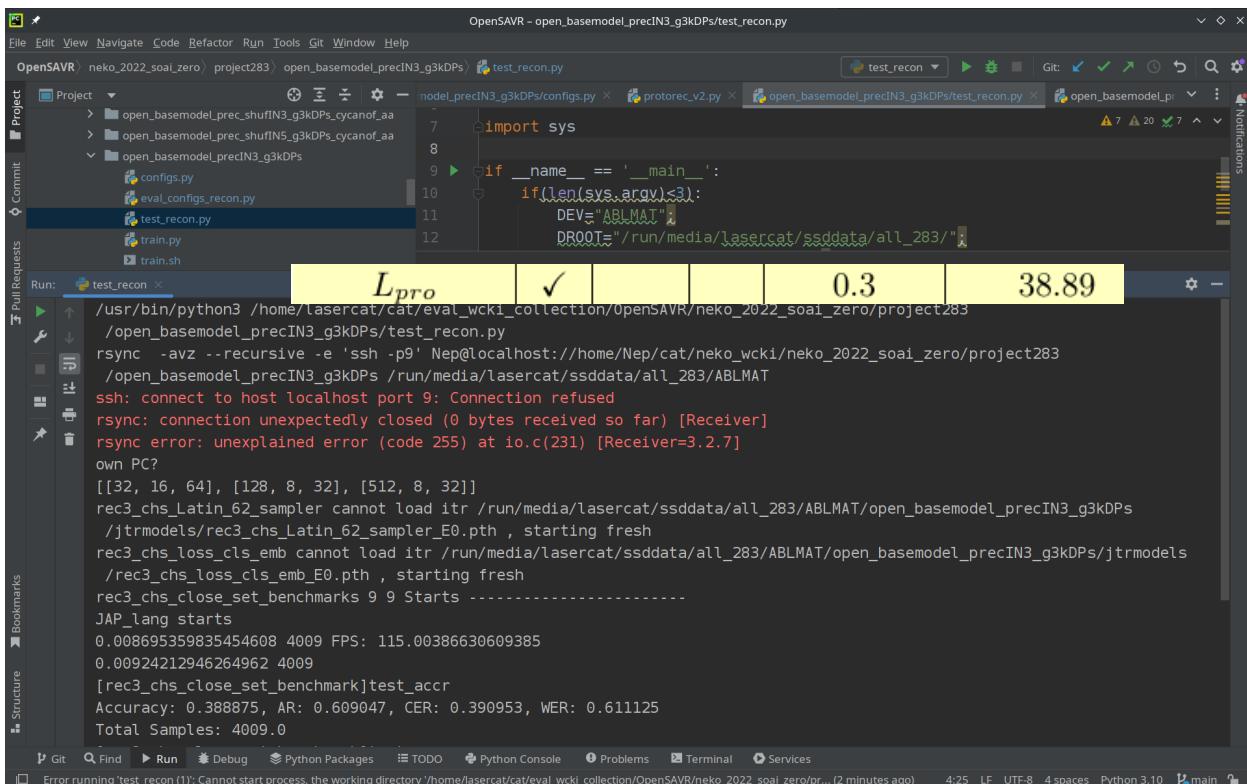
neko_2022_soai_zero/project283/open_basemodel/test_recon.py



```
Base - 36.89
/usr/bin/python3 /home/lasercat/cat/eval_wcki_collection/OpenSAVR/neko_2022_soai_zero/project283/open_basemodel/test_recon.py
rsync -avz --recursive -e 'ssh -p9' Nep@localhost://home/Nep/cat/neko_wcki/neko_2022_soai_zero/project283/open_basemodel
/home/lasercat/cat/eval_wcki_collection/opensavr-models/ABLMAT
ssh: connect to host localhost port 9: Connection refused
rsync: connection unexpectedly closed (0 bytes received so far) [Receiver]
rsync error: unexplained error (code 255) at io.c(231) [Receiver=3.2.7]
own PC?
[[32, 16, 64], [128, 8, 32], [512, 8, 32]]
base_chs_Latin_62_sampler cannot load itr /home/lasercat/cat/eval_wcki_collection/opensavr-models/ABLMAT/open_basemodel
/jtrmodels/base_chs_Latin_62_sampler_E0.pth , starting fresh
base_chs_loss_cls_emb cannot load itr /home/lasercat/cat/eval_wcki_collection/opensavr-models/ABLMAT/open_basemodel
/jtrmodels/base_chs_loss_cls_emb_E0.pth , starting fresh
base_chs_close_set_benchmarks 9 9 Starts -----
JAP_lang starts
0.005025506108798539 4009 FPS: 198.98493372622178
0.005666217025695938 4009
[base_chs_close_set_benchmark]test_accr
Accuracy: 0.368920, AR: 0.604191, CER: 0.395809, WER: 0.631080
Total Samples: 4009.0
[base chs close set benchmark]test accr
Externally added files can be added to Git
View Files Always Add Don't Ask Again
```

Step 2. Run Lpro:

neko_2022_soai_zero/project283/open_basemodel_precIN3_g3kDPs/test_recon.py



```
Lpro ✓ 0.3 38.89
/usr/bin/python3 /home/lasercat/cat/eval_wcki_collection/OpenSAVR/neko_2022_soai_zero/project283
/open_basemodel_precIN3_g3kDPs/test_recon.py
rsync -avz --recursive -e 'ssh -p9' Nep@localhost://home/Nep/cat/neko_wcki/neko_2022_soai_zero/project283
/open_basemodel_precIN3_g3kDPs /run/media/lasercat/ssddata/all_283/ABLMAT
ssh: connect to host localhost port 9: Connection refused
rsync: connection unexpectedly closed (0 bytes received so far) [Receiver]
rsync error: unexplained error (code 255) at io.c(231) [Receiver=3.2.7]
own PC?
[[32, 16, 64], [128, 8, 32], [512, 8, 32]]
rec3_chs_Latin_62_sampler cannot load itr /run/media/lasercat/ssddata/all_283/ABLMAT/open_basemodel_precIN3_g3kDPs
/jtrmodels/rec3_chs_Latin_62_sampler_E0.pth , starting fresh
rec3_chs_loss_cls_emb cannot load itr /run/media/lasercat/ssddata/all_283/ABLMAT/open_basemodel_precIN3_g3kDPs/jtrmodels
/rec3_chs_loss_cls_emb_E0.pth , starting fresh
rec3_chs_close_set_benchmarks 9 9 Starts -----
JAP_lang starts
0.008695359835454608 4009 FPS: 115.00386630609385
0.00924212946264962 4009
[rec3_chs_close_set_benchmark]test_accr
Accuracy: 0.388875, AR: 0.609047, CER: 0.390953, WER: 0.611125
Total Samples: 4009.0
Externally added files can be added to Git
View Files Always Add Don't Ask Again
```

We compared to a slightly stale version pre camera-ready version, so the number slightly improved a bit.

Step 3 Run Lpro+Lshuf

neko_2022_soai_zero/project283/open_basemodel_prec_shufIN3_g3kDPs/test_recon.py

```
missing debug info Lpro+Lshuf ✓ ✓ 0.3 40.71
missing debug info
0.0082994464297448 4009 FPS: 120.48998975449578
0.008841257435510093 4009
[shuf3_chs_close_set_benchmark]test_accr
Accuracy: 0.407084, AR: 0.621873, CER: 0.378127, WER: 0.592916
Total Samples: 4009.0
[shuf3_chs_close_set_benchmark]test_accr
Accuracy: 0.407084, AR: 0.621873, CER: 0.378127, WER: 0.592916
Total Samples: 4009.0
(0, {})
(0, {})
JAP_lang ends
-----
Process finished with exit code 0
```

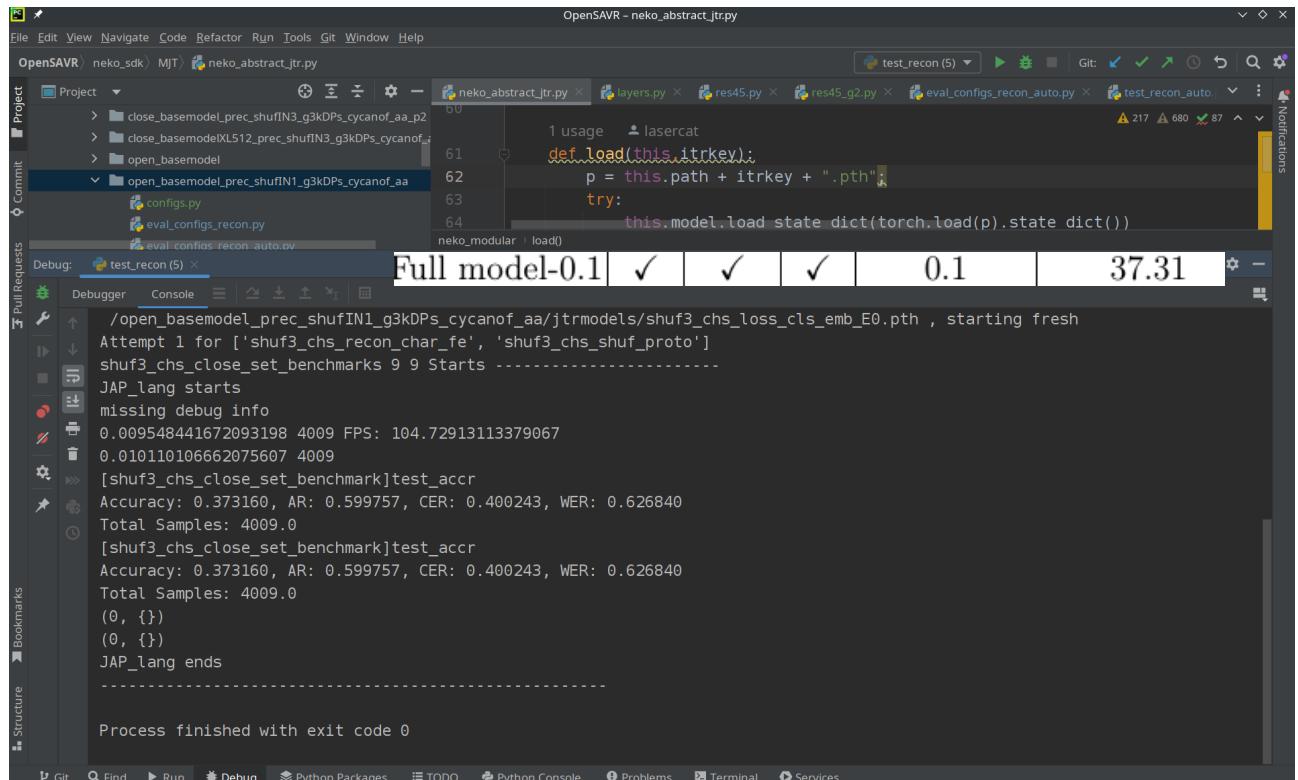
Step 4. Run full model

neko_2022_soai_zero/project283/open_basemodel_prec_shufIN3_g3kDPs_cycanoof_aa/test_recon.py

```
Attempt 1 for ['shuf3_chs_recon_char_fe', 'shuf3_chs_shuf_proto']
shuf3_chs_close_set_benchmarks 9 9 Starts -----
JAP_lang starts
missing debug info
missing debug info
missing debug info
0.008433302372404096 4009 FPS: 118.577510427262
0.008976204849056663 4009
[shuf3_chs_close_set_benchmark]test_accr
Accuracy: 0.409578, AR: 0.598385, CER: 0.401615, WER: 0.590422
Total Samples: 4009.0
[shuf3_chs_close_set_benchmark]test_accr
Accuracy: 0.409578, AR: 0.598385, CER: 0.401615, WER: 0.590422
Total Samples: 4009.0
(0, {})
(0, {})
JAP_lang ends
-----
Process finished with exit code 0
```

Step 5: Run Full model-0.1

neko_2022_soai_zero/project283/open_basemodel_prec_shufIN1_g3kDPs_cycanoof_aa/
test_recon_auto.py



OpenSAVR - neko_abstract_jtr.py

File Edit View Navigate Code Refactor Run Tools Git Window Help

OpenSAVR neko_sdk MJT neko_abstract_jtr.py

Project Commit Pull Requests Bookmarks Structure

Debug: test_recon (5) ✅ ✅ ✅ 0.1 37.31

Full model-0.1 | ✓ | ✓ | ✓ | 0.1 | 37.31

1 usage lasercat
def load(this,itrkey):
 p = this.path + itrkey + ".pth"
 try:
 this.model.load_state_dict(torch.load(p).state_dict())

neko_modular > load()

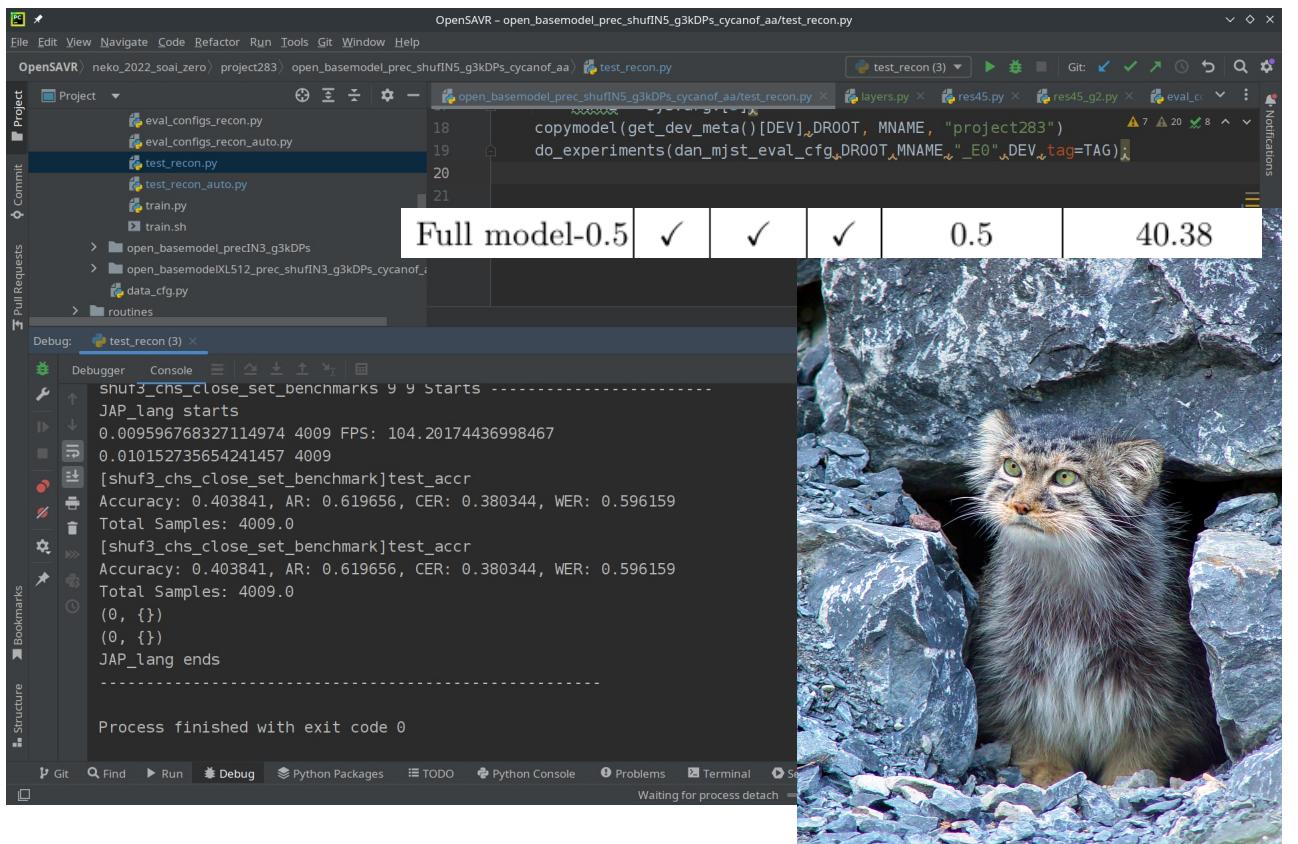
/open_basemodel_prec_shufIN1_g3kDPs_cycanoof_aa/jtrmodels/shuf3_chs_loss_cls_emb_E0.pth , starting fresh
Attempt 1 for ['shuf3_chs_recon_char_fe', 'shuf3_chs_shuf_proto']
shuf3_chs_close_set_benchmarks 9 9 Starts -----
JAP_lang starts
missing debug info
0.009548441672093198 4009 FPS: 104.72913113379067
0.010110106662075607 4009
[shuf3_chs_close_set_benchmark]test_accr
Accuracy: 0.373160, AR: 0.599757, CER: 0.400243, WER: 0.626840
Total Samples: 4009.0
[shuf3_chs_close_set_benchmark]test_accr
Accuracy: 0.373160, AR: 0.599757, CER: 0.400243, WER: 0.626840
Total Samples: 4009.0
(0, {})
(0, {})
JAP_lang ends

Process finished with exit code 0

Git Find Run Debug Python Packages TODO Python Console Problems Terminal Services

Step 6: Run Full model-0.5

neko_2022_soai_zero/project283/open_basemodel_prec_shufIN5_g3kDPs_cycanoof_aa/
test_recon.py



OpenSAVR - open_basemodel_prec_shufIN5_g3kDPs_cycanoof_aa/test_recon.py

File Edit View Navigate Code Refactor Run Tools Git Window Help

OpenSAVR neko_2022_soai_zero project283 open_basemodel_prec_shufIN5_g3kDPs_cycanoof_aa test_recon.py

Project Commit Pull Requests Bookmarks Structure

Debug: test_recon (3) ✅ ✅ ✅ 0.5 40.38

Full model-0.5 | ✓ | ✓ | ✓ | 0.5 | 40.38

copymodel(get_dev_meta() [DEV], DR00T, MNAME, "project283")
do_experiments(dan_mjst_eval_cfg, DR00T, MNAME, "E0", DEV, tag=TAG)

eval_configs_recon.py eval_configs_recon_auto.py test_recon.py test_recon_auto.py train.py train.sh open_basemodel_precIN3_g3kDPs open_basemodelXL512_prec_shufIN3_g3kDPs_cycanof_aa data_cfg.py routines

shuf3_chs_close_set_benchmarks 9 9 Starts -----
JAP_lang starts
0.009596768327114974 4009 FPS: 104.20174436998467
0.010152735654241457 4009
[shuf3_chs_close_set_benchmark]test_accr
Accuracy: 0.403841, AR: 0.619656, CER: 0.380344, WER: 0.596159
Total Samples: 4009.0
[shuf3_chs_close_set_benchmark]test_accr
Accuracy: 0.403841, AR: 0.619656, CER: 0.380344, WER: 0.596159
Total Samples: 4009.0
(0, {})
(0, {})
JAP_lang ends

Process finished with exit code 0

Git Find Run Debug Python Packages TODO Python Console Problems Terminal Services

Waiting for process detach



3. Running the Open-Set benchmark

Step 1: Run the regular model:

```
neko_2022_soai_zero/project283/open_basemodel_prec_shufIN3_g3kDPs_cyanof_aa/  
test_recon_auto.py
```

OSR

Latin	<i>Kana</i>	Ours	75.08	56.38	96.95	71.29
-------	-------------	------	-------	-------	-------	-------

```
[shuf3_chs_close_set_benchmark]test_accr  
KACR: 0.750776, URCL: 0.563774, UPRE 0.969474, F 0.712949
```

GZSL

		Ours	40.96	-	-	-
--	--	------	-------	---	---	---

```
[shuf3_chs_close_set_benchmark]test_accr  
Accuracy: 0.409578, AR: 0.598385, CER: 0.401615, WER: 0.590422  
Total Samples: 4009.0
```

GOSR

<i>Unique Kanji</i>	Ours	68.43	34.23	80.58	48.05
---------------------	------	-------	-------	-------	-------

```
[shuf3_chs_close_set_benchmark]test_accr  
KACR: 0.684329, URCL: 0.342312, UPRE 0.805812, F 0.480504
```

OSTR

Ours	71.86	69.72	90.86	78.90
------	-------	-------	-------	-------

```
[shuf3_chs_close_set_benchmark]test_accr  
KACR: 0.718619, URCL: 0.697186, UPRE 0.908639, F 0.788991
```

Step 2: Run the Large model:

neko_2022_soai_zero/project283/open_basemodelXL512_prec_shufIN3_g3kDPs_cycanof_aa/test_recon_auto.py

OSR

		Ours-Large	78.49	58.81	97.33	73.32
[shuf3_chs_close_set_benchmark]test_accr						
KACR: 0.784902, URCL: 0.588100, UPRE 0.973341, F 0.733197						

GZSL

Latin	Ours-Large	69.29	41.19	85.05	55.50
[shuf3_chs_close_set_benchmark]test_accr					
Accuracy: 0.425792, AR: 0.630529, CER: 0.369471, WER: 0.574208					

GOSR

Latin	Ours-Large	69.29	41.19	85.05	55.50
[shuf3_chs_close_set_benchmark]test_accr					
KACR: 0.692860, URCL: 0.411897, UPRE 0.850521, F 0.555009					

OSTR

	Ours-Large	72.33	72.96	92.62	81.62
[shuf3_chs_close_set_benchmark]test_accr					
KACR: 0.723326, URCL: 0.729614, UPRE 0.926150, F 0.816218					

Mhew... That's so much for
openset.



4. Running the Close-Set benchmark

Step 1: Run the regular model

neko_2022_soai_zero/project283/close_basemodel_prec_shufIN3_g3kDPs_cycanof_aa_p2/test_recon.py

I am running on a 1070 writing the docu so the speed is slower.

The p40 station is back in my den, um, I mean home. The lockdown ends so I am back to my post.

Step 2: Run the large model:

neko_2022_soai_zero/project283/close_basemodelXL512_prec_shufIN3_g3kDPs_cycanof_aa_p3/test_recon.py

Ours-Large	-	92.33	84.38	85.16	91.58	90.64	Tesla P40	12	54
CUTE starts									
0.025062113172478147 288 FPS: 39.90086522704501									
0.03063066966003842 288									
[shuf3_mjst_close_set_benchmark]test_accr									
Accuracy: 0.843750, AR: 0.921003, CER: 0.078997, WER: 0.156250									
Total Samples: 288.0									
IIIT5k starts									
0.02430103898048401 3000 FPS: 41.150503927140434									
0.02430763498942057 3000									
[shuf3_mjst_close_set_benchmark]test_accr									
Accuracy: 0.923333, AR: 0.971249, CER: 0.028751, WER: 0.076667									
SVT starts									
0.02618659187505566 647 FPS: 38.18748177583817									
0.026217802967501938 647									
[shuf3_mjst_close_set_benchmark]test_accr									
Accuracy: 0.851623, AR: 0.948103, CER: 0.051897, WER: 0.148377									
IC03 starts									
0.027263230243493666 867 FPS: 36.679439342616									
0.027286186625250758 867									
[shuf3_mjst_close_set_benchmark]test_accr									
Accuracy: 0.915802, AR: 0.968820, CER: 0.031180, WER: 0.084198									
IC13 starts									
0.026337225448909065 1015 FPS: 37.96907164499447									
0.026356897448084036 1015									
[shuf3_mjst_close_set_benchmark]test_accr									
Accuracy: 0.906404, AR: 0.970692, CER: 0.029308, WER: 0.093596									

So the speed is not much lower, why? This is the nature of single batched test. You may set a larger batchsize, and you will find out that this babe can hit 100+/80+ FPS on a mere 1070. But heck we need to be fair when comparing, so in the paper we report single-batched speed.



5. Experiments on Close-set Impacts.

6. Training models

7. Happy International Manuls' Day