Download zip from <https://github.com/scpd-proed/XCS224N-A3>

Unzip content into

C:\Users\ADMIN\Documents\Stanford\NLP\_DeepLearning\Assignment3\XCS224N-A3-master

* Open Anaconda Prompt

(base) C:\Users\ADMIN>python --version

Python 3.7.3

* navigate to your project

(base) C:\Users\ADMIN>cd C:\Users\ADMIN\Documents\Stanford\NLP\_DeepLearning\Assignment3\XCS224N-A3-master

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* virtual environment created using virtualenv failed to install pytorch!
* virtual environment created using conda succeeded in installing pytorch!

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# Environment Setup Start

* Create a virtual environment using conda

(base) C:\Users\ADMIN\Documents\Stanford\NLP\_DeepLearning\Assignment3\XCS224N-A3-master>conda env create --file local\_env.yml

|  |
| --- |
| Collecting package metadata (repodata.json): done  Solving environment: done  Downloading and Extracting Packages  certifi-2019.11.28 | 154 KB | #################################### | 100%  mkl\_random-1.1.0 | 233 KB | #################################### | 100%  mkl-service-2.3.0 | 210 KB | #################################### | 100%  setuptools-44.0.0 | 527 KB | #################################### | 100%  wheel-0.33.6 | 58 KB | #################################### | 100%  wincertstore-0.2 | 14 KB | #################################### | 100%  six-1.13.0 | 27 KB | #################################### | 100%  pip-19.3.1 | 1.7 MB | #################################### | 100%  python-3.6.10 | 15.9 MB | #################################### | 100%  tqdm-4.19.9 | 72 KB | #################################### | 100%  sqlite-3.30.1 | 627 KB | #################################### | 100%  mkl\_fft-1.0.15 | 118 KB | #################################### | 100%  vs2015\_runtime-14.16 | 1.1 MB | #################################### | 100%  Preparing transaction: done  Verifying transaction: done  Executing transaction: done  #  # To activate this environment, use  #  # $ conda activate A3  #  # To deactivate an active environment, use  #  # $ conda deactivate |

Note: The conda env is located in C:\ProgramData\Anaconda3\envs\A3.

* Activate the virtual environment using conda

(base) C:\Users\ADMIN\Documents\Stanford\NLP\_DeepLearning\Assignment3\XCS224N-A3-master>conda activate A3

|  |
| --- |
| (A3) C:\Users\ADMIN\Documents\Stanford\NLP\_DeepLearning\Assignment3\XCS224N-A3-master> |

* install PyTorch 1.0.0 or above with the CUDA option set to None. (do not require CUDA)

|  |
| --- |
| <https://pytorch.org/get-started/locally/> |

(A3) C:\Users\ADMIN\Documents\Stanford\NLP\_DeepLearning\Assignment3\XCS224N-A3-master>conda install pytorch torchvision cpuonly -c pytorch

|  |
| --- |
| Collecting package metadata (current\_repodata.json): done  Solving environment: done  ## Package Plan ##  environment location: C:\ProgramData\Anaconda3\envs\A3  added / updated specs:  - cpuonly  - pytorch  - torchvision  The following packages will be downloaded:  package | build  ---------------------------|-----------------  cffi-1.13.2 | py36h7a1dbc1\_0 223 KB  freetype-2.9.1 | ha9979f8\_1 450 KB  jpeg-9b | hb83a4c4\_2 245 KB  libpng-1.6.37 | h2a8f88b\_0 333 KB  libtiff-4.1.0 | h56a325e\_0 737 KB  ninja-1.9.0 | py36h74a9793\_0 238 KB  olefile-0.46 | py36\_0 49 KB  pillow-7.0.0 | py36hcc1f983\_0 652 KB  pycparser-2.19 | py36\_0 171 KB  pytorch-1.4.0 | py3.6\_cpu\_0 61.9 MB pytorch  tk-8.6.8 | hfa6e2cd\_0 3.1 MB  torchvision-0.5.0 | py36\_cpu 5.8 MB pytorch  xz-5.2.4 | h2fa13f4\_4 458 KB  zlib-1.2.11 | h62dcd97\_3 110 KB  zstd-1.3.7 | h508b16e\_0 337 KB  ------------------------------------------------------------  Total: 74.7 MB  The following NEW packages will be INSTALLED:  cffi pkgs/main/win-64::cffi-1.13.2-py36h7a1dbc1\_0  cpuonly pytorch/noarch::cpuonly-1.0-0  freetype pkgs/main/win-64::freetype-2.9.1-ha9979f8\_1  jpeg pkgs/main/win-64::jpeg-9b-hb83a4c4\_2  libpng pkgs/main/win-64::libpng-1.6.37-h2a8f88b\_0  libtiff pkgs/main/win-64::libtiff-4.1.0-h56a325e\_0  ninja pkgs/main/win-64::ninja-1.9.0-py36h74a9793\_0  olefile pkgs/main/win-64::olefile-0.46-py36\_0  pillow pkgs/main/win-64::pillow-7.0.0-py36hcc1f983\_0  pycparser pkgs/main/win-64::pycparser-2.19-py36\_0  pytorch pytorch/win-64::pytorch-1.4.0-py3.6\_cpu\_0  tk pkgs/main/win-64::tk-8.6.8-hfa6e2cd\_0  torchvision pytorch/win-64::torchvision-0.5.0-py36\_cpu  xz pkgs/main/win-64::xz-5.2.4-h2fa13f4\_4  zlib pkgs/main/win-64::zlib-1.2.11-h62dcd97\_3  zstd pkgs/main/win-64::zstd-1.3.7-h508b16e\_0  Proceed ([y]/n)? y  Downloading and Extracting Packages  cffi-1.13.2 | 223 KB | #################################### | 100%  olefile-0.46 | 49 KB | #################################### | 100%  pillow-7.0.0 | 652 KB | #################################### | 100%  libtiff-4.1.0 | 737 KB | #################################### | 100%  libpng-1.6.37 | 333 KB | #################################### | 100%  zstd-1.3.7 | 337 KB | #################################### | 100%  xz-5.2.4 | 458 KB | #################################### | 100%  pycparser-2.19 | 171 KB | #################################### | 100%  torchvision-0.5.0 | 5.8 MB | #################################### | 100%  ninja-1.9.0 | 238 KB | #################################### | 100%  pytorch-1.4.0 | 61.9 MB | #################################### | 100%  zlib-1.2.11 | 110 KB | #################################### | 100%  jpeg-9b | 245 KB | #################################### | 100%  freetype-2.9.1 | 450 KB | #################################### | 100%  tk-8.6.8 | 3.1 MB | #################################### | 100%  Preparing transaction: done  Verifying transaction: done  Executing transaction: done |

* To ensure that PyTorch was installed correctly, we can verify the installation by running sample PyTorch code. Here we will construct a randomly initialized tensor.

|  |
| --- |
| (A3) C:\Users\ADMIN\Documents\Stanford\NLP\_DeepLearning\Assignment3\XCS224N-A3-master>python  Python 3.6.10 |Anaconda, Inc.| (default, Jan 7 2020, 15:18:16) [MSC v.1916 64 bit (AMD64)] on win32  Type "help", "copyright", "credits" or "license" for more information.  >>> from \_\_future\_\_ import print\_function  >>> import torch  >>> x = torch.rand(5, 3)  >>> print(x)  tensor([[0.3161, 0.6697, 0.7309],  [0.3820, 0.8877, 0.0812],  [0.1249, 0.8654, 0.1165],  [0.2406, 0.7663, 0.6510],  [0.7771, 0.3377, 0.6710]])  >>> torch.cuda.is\_available()  False  >>> exit() |

* Check all the python packages in this virtual environment

|  |
| --- |
| (A3) C:\Users\ADMIN\Documents\Stanford\NLP\_DeepLearning\Assignment3\XCS224N-A3-master>conda list  # packages in environment at C:\ProgramData\Anaconda3\envs\A3:  #  # Name Version Build Channel  blas 1.0 mkl  certifi 2019.11.28 py36\_0  cffi 1.13.2 py36h7a1dbc1\_0  cpuonly 1.0 0 pytorch  freetype 2.9.1 ha9979f8\_1  icc\_rt 2019.0.0 h0cc432a\_1  intel-openmp 2019.4 245  jpeg 9b hb83a4c4\_2  libpng 1.6.37 h2a8f88b\_0  libtiff 4.1.0 h56a325e\_0  mkl 2019.4 245  mkl-service 2.3.0 py36hb782905\_0  mkl\_fft 1.0.15 py36h14836fe\_0  mkl\_random 1.1.0 py36h675688f\_0  ninja 1.9.0 py36h74a9793\_0  numpy 1.16.4 py36h19fb1c0\_0  numpy-base 1.16.4 py36hc3f5095\_0  olefile 0.46 py36\_0  pillow 7.0.0 py36hcc1f983\_0  pip 19.3.1 py36\_0  pycparser 2.19 py36\_0  python 3.6.10 h9f7ef89\_0  pytorch 1.4.0 py3.6\_cpu\_0 [cpuonly] pytorch  setuptools 44.0.0 py36\_0  six 1.13.0 py36\_0  sqlite 3.30.1 he774522\_0  tk 8.6.8 hfa6e2cd\_0  torchvision 0.5.0 py36\_cpu [cpuonly] pytorch  tqdm 4.19.9 py36\_0  vc 14.1 h0510ff6\_4  vs2015\_runtime 14.16.27012 hf0eaf9b\_1  wheel 0.33.6 py36\_0  wincertstore 0.2 py36h7fe50ca\_0  xz 5.2.4 h2fa13f4\_4  zlib 1.2.11 h62dcd97\_3  zstd 1.3.7 h508b16e\_0 |

# Environment Setup End

# Coding/Testing Start

1. (6 points) Implement the \_\_init\_\_ and parse\_step functions in the PartialParse class in parser\_transitions.py. This implements the transition mechanics your parser will use. You can run basic (non-exhaustive) tests by running python parser\_transitions.py. (Note: You will find the parser\_transitions.py file inside the utils folder)

* Before coding test

|  |
| --- |
| (A3) C:\Users\ADMIN\Documents\Stanford\NLP\_DeepLearning\Assignment3\XCS224N-A3-master>cd utils  (A3) C:\Users\ADMIN\Documents\Stanford\NLP\_DeepLearning\Assignment3\XCS224N-A3-master\utils>python parser\_transitions.py  Traceback (most recent call last):  File "parser\_transitions.py", line 189, in <module>  test\_parse\_step()  File "parser\_transitions.py", line 128, in test\_parse\_step  ("ROOT", "the", "cat"), ("sat",), ())  File "parser\_transitions.py", line 115, in test\_step  "{:} test resulted in stack {:}, expected {:}".format(name, stack, ex\_stack)  AssertionError: SHIFT test resulted in stack ('ROOT', 'the'), expected ('ROOT', 'the', 'cat') |

* After coding test

|  |
| --- |
| (A3) C:\Users\ADMIN\Documents\Stanford\NLP\_DeepLearning\Assignment3\XCS224N-A3-master\utils>python parser\_transitions.py  SHIFT test passed!  LEFT-ARC test passed!  RIGHT-ARC test passed!  parse test passed!  Traceback (most recent call last):  File "parser\_transitions.py", line 200, in <module>  test\_minibatch\_parse()  File "parser\_transitions.py", line 185, in test\_minibatch\_parse  deps = minibatch\_parse(sentences, DummyModel(), 2)  File "parser\_transitions.py", line 113, in minibatch\_parse  return dependencies  NameError: name 'dependencies' is not defined |

1. (6 points) Implement this algorithm in the minibatch\_parse function in parser\_transitions.py. You can run basic (non-exhaustive) tests by running python parser transitions.py.

* After coding test

|  |
| --- |
| (A3) C:\Users\ADMIN\Documents\Stanford\NLP\_DeepLearning\Assignment3\XCS224N-A3-master\utils>python parser\_transitions.py  SHIFT test passed!  LEFT-ARC test passed!  RIGHT-ARC test passed!  parse test passed!  minibatch\_parse test passed! |

1. (9 points) In parser\_model.py you will find skeleton code to implement this simple neural net-

work using PyTorch. Complete the \_\_init\_\_ , embedding\_lookup and forward functions to implement the model. Then complete the train\_for\_epoch function within the run.py file.

* Debug Run (set debug=True in the main function of run.py)

|  |
| --- |
| (A3) C:\Users\ADMIN\Documents\Stanford\NLP\_DeepLearning\Assignment3\XCS224N-A3-master\utils>cd ..  (A3) C:\Users\ADMIN\Documents\Stanford\NLP\_DeepLearning\Assignment3\XCS224N-A3-master>python run.py  ================================================================================  INITIALIZING  ================================================================================  Loading data...  took 1.24 seconds  Building parser...  took 0.02 seconds  Loading pretrained embeddings...  took 1.89 seconds  Vectorizing data...  took 0.04 seconds  Preprocessing training data...  took 0.74 seconds  took 0.02 seconds  ================================================================================  TRAINING  ================================================================================  Epoch 1 out of 10  100%|█████████████████████████████████████████████████████████████████| 48/48 [00:05<00:00, 9.41it/s]  Average Train Loss: 0.6030602740744749  Evaluating on dev set  125250it [00:00, 20876513.11it/s]  - dev UAS: 53.75  New best dev UAS! Saving model.  Epoch 2 out of 10  100%|█████████████████████████████████████████████████████████████████| 48/48 [00:05<00:00, 9.16it/s]  Average Train Loss: 0.3383739236742258  Evaluating on dev set  125250it [00:00, 12527102.63it/s]  - dev UAS: 59.95  New best dev UAS! Saving model.  Epoch 3 out of 10  100%|█████████████████████████████████████████████████████████████████| 48/48 [00:05<00:00, 9.35it/s]  Average Train Loss: 0.2726467742274205  Evaluating on dev set  125250it [00:00, 9635667.20it/s]  - dev UAS: 63.48  New best dev UAS! Saving model.  Epoch 4 out of 10  100%|█████████████████████████████████████████████████████████████████| 48/48 [00:05<00:00, 9.30it/s]  Average Train Loss: 0.2405658019706607  Evaluating on dev set  125250it [00:00, 20881492.01it/s]  - dev UAS: 65.54  New best dev UAS! Saving model.  Epoch 5 out of 10  100%|█████████████████████████████████████████████████████████████████| 48/48 [00:05<00:00, 9.27it/s]  Average Train Loss: 0.21522947711249193  Evaluating on dev set  125250it [00:00, 20876513.11it/s]  - dev UAS: 66.95  New best dev UAS! Saving model.  Epoch 6 out of 10  100%|█████████████████████████████████████████████████████████████████| 48/48 [00:05<00:00, 9.24it/s]  Average Train Loss: 0.19255619713415703  Evaluating on dev set  125250it [00:00, 9635490.47it/s]  - dev UAS: 69.23  New best dev UAS! Saving model.  Epoch 7 out of 10  100%|█████████████████████████████████████████████████████████████████| 48/48 [00:05<00:00, 9.23it/s]  Average Train Loss: 0.17624581884592772  Evaluating on dev set  125250it [00:00, 9635843.94it/s]  - dev UAS: 69.51  New best dev UAS! Saving model.  Epoch 8 out of 10  100%|█████████████████████████████████████████████████████████████████| 48/48 [00:05<00:00, 10.36it/s]  Average Train Loss: 0.161817976894478  Evaluating on dev set  125250it [00:00, 20876513.11it/s]  - dev UAS: 68.30  Epoch 9 out of 10  100%|█████████████████████████████████████████████████████████████████| 48/48 [00:05<00:00, 9.31it/s]  Average Train Loss: 0.14961245702579618  Evaluating on dev set  125250it [00:00, 20877342.77it/s]  - dev UAS: 70.46  New best dev UAS! Saving model.  Epoch 10 out of 10  100%|█████████████████████████████████████████████████████████████████| 48/48 [00:05<00:00, 10.29it/s]  Average Train Loss: 0.1396813876926899  Evaluating on dev set  125250it [00:00, 9635843.94it/s]  - dev UAS: 72.45  New best dev UAS! Saving model. |

* Full Training Run (set debug=False in the main function of run.py)

|  |
| --- |
| (A3) C:\Users\ADMIN\Documents\Stanford\NLP\_DeepLearning\Assignment3\XCS224N-A3-master>python run.py  ================================================================================  INITIALIZING  ================================================================================  Loading data...  took 1.17 seconds  Building parser...  took 0.71 seconds  Loading pretrained embeddings...  took 1.91 seconds  Vectorizing data...  took 0.94 seconds  Preprocessing training data...  took 25.95 seconds  took 0.01 seconds  ================================================================================  TRAINING  ================================================================================  Epoch 1 out of 10  100%|█████████████████████████████████████████████████████████████| 1848/1848 [03:48<00:00, 8.32it/s]  Average Train Loss: 0.18281215089030597  Evaluating on dev set  1445850it [00:00, 38049293.44it/s]  - dev UAS: 84.33  New best dev UAS! Saving model.  Epoch 2 out of 10  100%|█████████████████████████████████████████████████████████████| 1848/1848 [03:48<00:00, 7.90it/s]  Average Train Loss: 0.11587165739564668  Evaluating on dev set  1445850it [00:00, 45181374.43it/s]  - dev UAS: 86.10  New best dev UAS! Saving model.  Epoch 3 out of 10  100%|█████████████████████████████████████████████████████████████| 1848/1848 [03:47<00:00, 7.86it/s]  Average Train Loss: 0.10156061309208343  Evaluating on dev set  1445850it [00:00, 76097154.52it/s]  - dev UAS: 87.03  New best dev UAS! Saving model.  Epoch 4 out of 10  100%|█████████████████████████████████████████████████████████████| 1848/1848 [03:48<00:00, 8.50it/s]  Average Train Loss: 0.09255004554051599  Evaluating on dev set  1445850it [00:00, 45184067.52it/s]  - dev UAS: 87.46  New best dev UAS! Saving model.  Epoch 5 out of 10  100%|█████████████████████████████████████████████████████████████| 1848/1848 [03:47<00:00, 8.25it/s]  Average Train Loss: 0.08629336989045272  Evaluating on dev set  1445850it [00:00, 55610076.37it/s]  - dev UAS: 87.63  New best dev UAS! Saving model.  Epoch 6 out of 10  100%|█████████████████████████████████████████████████████████████| 1848/1848 [03:47<00:00, 8.75it/s]  Average Train Loss: 0.08131379588382133  Evaluating on dev set  1445850it [00:00, 38047861.11it/s]  - dev UAS: 88.17  New best dev UAS! Saving model.  Epoch 7 out of 10  100%|█████████████████████████████████████████████████████████████| 1848/1848 [03:45<00:00, 8.29it/s]  Average Train Loss: 0.07701538577151698  Evaluating on dev set  1445850it [00:00, 72296878.18it/s]  - dev UAS: 88.44  New best dev UAS! Saving model.  Epoch 8 out of 10  100%|█████████████████████████████████████████████████████████████| 1848/1848 [03:46<00:00, 7.94it/s]  Average Train Loss: 0.07350902506291415  Evaluating on dev set  1445850it [00:00, 48193130.94it/s]  - dev UAS: 88.52  New best dev UAS! Saving model.  Epoch 9 out of 10  100%|█████████████████████████████████████████████████████████████| 1848/1848 [03:45<00:00, 8.59it/s]  Average Train Loss: 0.06984522930866519  Evaluating on dev set  1445850it [00:00, 76096199.65it/s]  - dev UAS: 88.47  Epoch 10 out of 10  100%|█████████████████████████████████████████████████████████████| 1848/1848 [03:46<00:00, 7.88it/s]  Average Train Loss: 0.06716338053418251  Evaluating on dev set  1445850it [00:00, 49858050.83it/s]  - dev UAS: 88.65  New best dev UAS! Saving model.  ================================================================================  TESTING  ================================================================================  Restoring the best model weights found on the dev set  Final evaluation on test set  2919736it [00:00, 108143343.70it/s]  - test UAS: 88.87  Done! |

* Zip all .py files and include all files under ./data and ./utils and ./results folders to produce assignment3.zip