## Theory Homework

1. What are the key architectural features that make these systems suitable for AI workloads?
   =>
   - Massively parallel, to handle high amount of FLOPs and parallel nature of neural network.
   - I/O overhead minimization is done by large on-chip memory and high-bendwidth memory or WSE architecture.
   - Spatial architecture for AI (neural network) workload to map each portion of network to separate dedicated computation unit and high fast interconnect among the units.
   - Workload specialization, e.g. matrix multiplication.

2. Identify the primary differences between these AI accelerator systems in terms of their architecture and programming models.
   => Followings are primary characteristics of described accelerators which makes them stand out
   - Cerebras: WSE architecture, very high general purpose #cores (850000) in single wafer for fast interconnect
   - GraphCore: MIMD architecture, bulk-synchronous parallel model of executing task, focussed on matrix multiplication
   - Sambanova: Reconfigurable SIMD pipeline, Spatial architecture
   - GROQ: Specialized for inference with batch size 1, specialized SIMD unit for matrix, vector and data shaping operations, no memory hierarchy

3. Based on hands-on sessions, describe a typical workflow for refactoring an AI model to run on one of ALCF's AI testbeds (e.g., SambaNova or Cerebras). What tools or software stacks are typically used in this process?
   =>
   For Cerebras the hardware effect is transparent at PyTorch script level.
   - Special PyTorch build is used (needs installation when setting up runtime environment) to run PyTorch model script
   - For Cerebras H/W acceleration, custom graph compiler is used. Kernels functions are device specific and the PyTorch build provide placement and routing logic.

   For Sambanova model gets compiled with custom command but the script stays unchanged

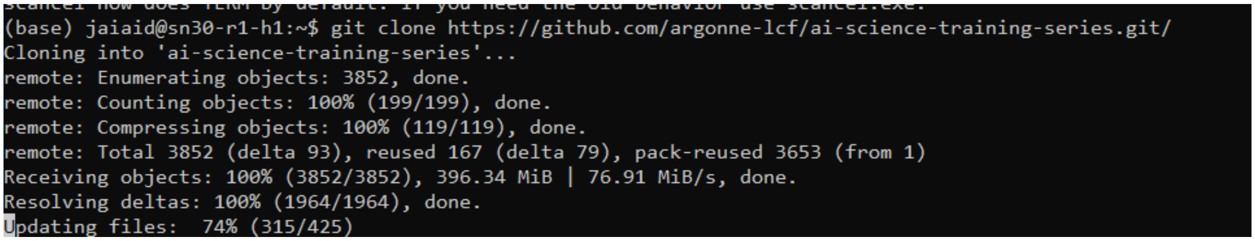
4. Give an example of a project that would benefit from AI accelerators and why?
   =>
   A multitask agent supporting multiple people’s work in same project.
   - LLM fine-tune and inference to assist project members. Multiple member means high computation needs

- Assistance in organizing workspace. E.g. propose directory structure in project PC
- Visual data analysis or real time observation of sample(s)

The above mainly will need high simultaneous computation need due to jobs submitted from multiple member PC at high frequency.

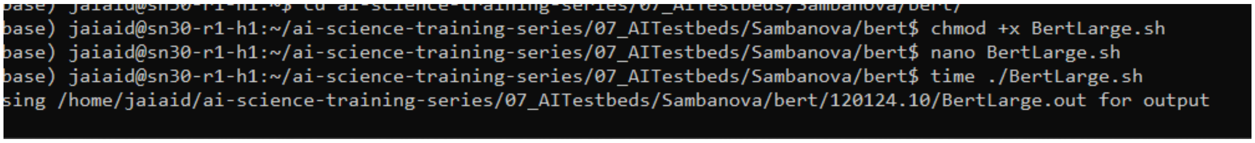
## Sambanova Workflow

1. Login and clone the repo

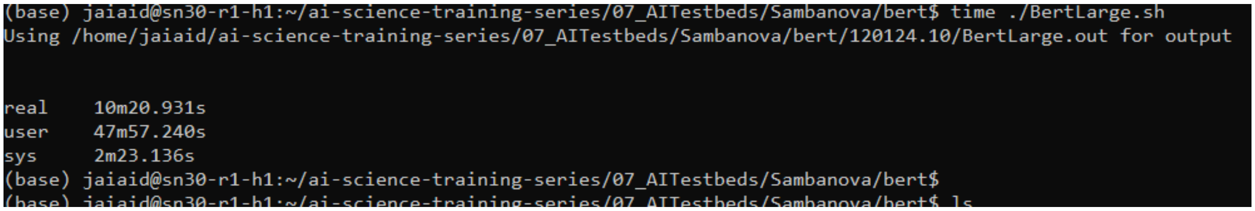


```
scancel now does TERM by default. If you need the old behavior use scancel.exe.
(base) jaiaid@sn30-r1-h1:~$ git clone https://github.com/argonne-lcf/ai-science-training-series.git/
Cloning into 'ai-science-training-series'...
remote: Enumerating objects: 3852, done.
remote: Counting objects: 100% (199/199), done.
remote: Compressing objects: 100% (119/119), done.
remote: Total 3852 (delta 93), reused 167 (delta 79), pack-reused 3653 (from 1)
Receiving objects: 100% (3852/3852), 396.34 MiB | 76.91 MiB/s, done.
Resolving deltas: 100% (1964/1964), done.
Updating files:  74% (315/425)
```

2. Edit task no. and run



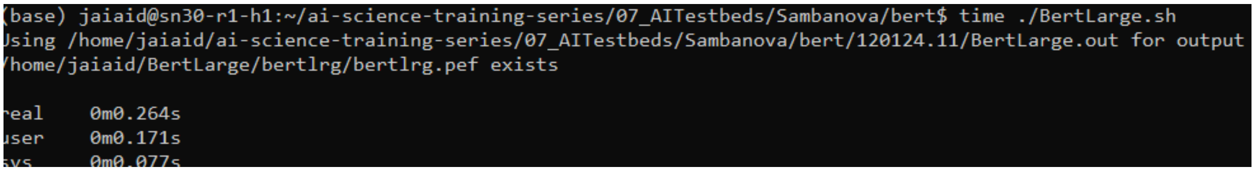
```
base) jaiaid@sn30-r1-h1:~$ cd ai-science-training-series/07_AITestbeds/Sambanova/bert/
base) jaiaid@sn30-r1-h1:~/ai-science-training-series/07_AITestbeds/Sambanova/bert$ chmod +x BertLarge.sh
base) jaiaid@sn30-r1-h1:~/ai-science-training-series/07_AITestbeds/Sambanova/bert$ nano BertLarge.sh
base) jaiaid@sn30-r1-h1:~/ai-science-training-series/07_AITestbeds/Sambanova/bert$ time ./BertLarge.sh
sing /home/jaiaid/ai-science-training-series/07_AITestbeds/Sambanova/bert/120124.10/BertLarge.out for output
```



```
(base) jaiaid@sn30-r1-h1:~/ai-science-training-series/07_AITestbeds/Sambanova/bert$ time ./BertLarge.sh
Using /home/jaiaid/ai-science-training-series/07_AITestbeds/Sambanova/bert/120124.10/BertLarge.out for output


real    10m20.931s
user    47m57.240s
sys     2m23.136s
(base) jaiaid@sn30-r1-h1:~/ai-science-training-series/07_AITestbeds/Sambanova/bert$
(base) jaiaid@sn30-r1-h1:~/ai-science-training-series/07_AITestbeds/Sambanova/bert$ ls
```

After compilation of .pef



```
(base) jaiaid@sn30-r1-h1:~/ai-science-training-series/07_AITestbeds/Sambanova/bert$ time ./BertLarge.sh
Jsing /home/jaiaid/ai-science-training-series/07_AITestbeds/Sambanova/bert/120124.11/BertLarge.out for output
/home/jaiaid/BertLarge/bertlrg/bertlrg.pef exists

real    0m0.264s
user    0m0.171s
sys     0m0.077s
```

| ntasks | 1 | 2 | 4 | 8 | 16 | 32 |
|---|---|---|---|---|---|---|
| time(sec.) | 0.264 | 0.267 | 0.263 | 0.265 | 0.263 | 0.158 |

The scaling reason is not clear. If we keep ntasks at 16 but varies –ntasks-per-node to value other than 16 (1, 8, 32 are tried) the execution time reduced to ~0.16 second.

The logs can be found at
`/home/jaiaid/ai-science-training-series/07_AITestbeds/Sambanova/bert`

The *.pef can be found at

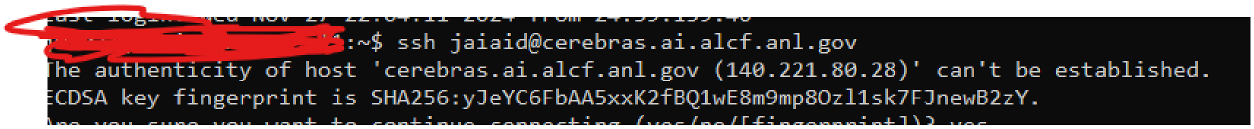
`/home/jaiaid/BertLarge/bertlrg`

## Cerebras Node Workflow

We faced some issue with Cerebrus and Groq. Possibly due to late submission (Dataset and job not found)

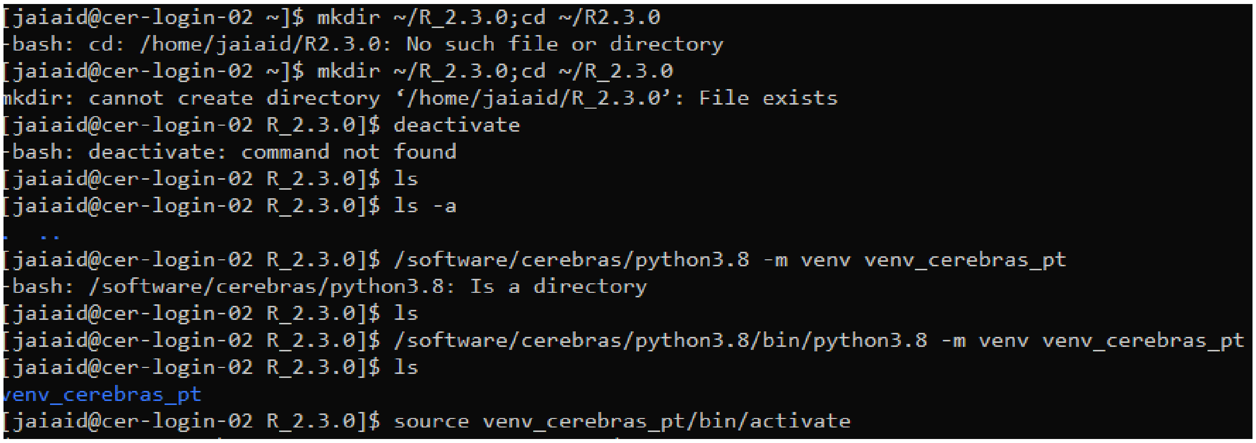
**Env. Preparation**

1. Login to Cerebras

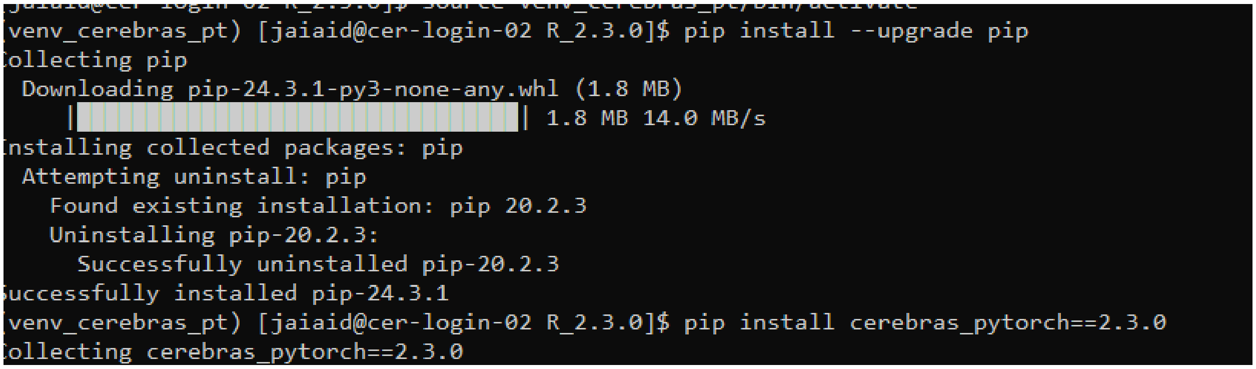


```
[redacted]:~$ ssh jaiaid@cerebras.ai.alcf.anl.gov
The authenticity of host 'cerebras.ai.alcf.anl.gov (140.221.80.28)' can't be established.
ECDSA key fingerprint is SHA256:yJeYC6FbAA5xxK2fBQ1wE8m9mp8Ozl1sk7FJnewB2zY.
```

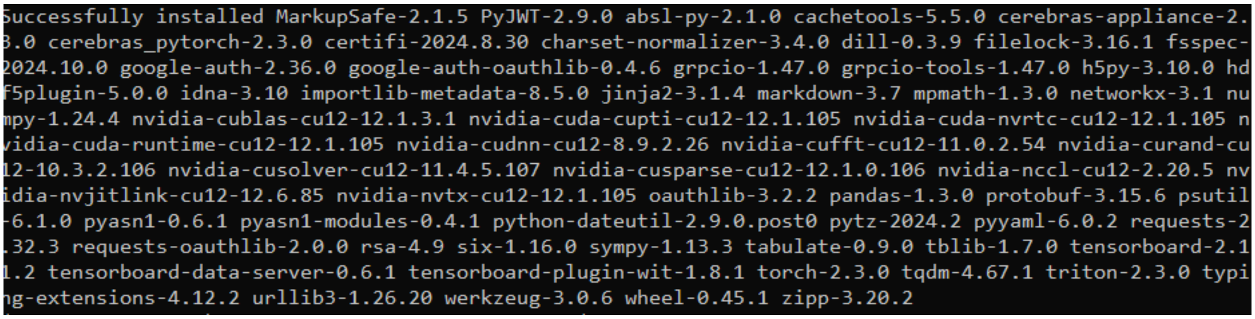
2. Create the virtual environment



```
[jaiaid@cer-login-02 ~]$ mkdir ~/R_2.3.0;cd ~/R2.3.0
-bash: cd: /home/jaiaid/R2.3.0: No such file or directory
[jaiaid@cer-login-02 ~]$ mkdir ~/R_2.3.0;cd ~/R_2.3.0
mkdir: cannot create directory ‘/home/jaiaid/R_2.3.0’: File exists
[jaiaid@cer-login-02 R_2.3.0]$ deactivate
-bash: deactivate: command not found
[jaiaid@cer-login-02 R_2.3.0]$ ls
[jaiaid@cer-login-02 R_2.3.0]$ ls -a
.  ..
[jaiaid@cer-login-02 R_2.3.0]$ /software/cerebras/python3.8 -m venv venv_cerebras_pt
-bash: /software/cerebras/python3.8: Is a directory
[jaiaid@cer-login-02 R_2.3.0]$ ls
[jaiaid@cer-login-02 R_2.3.0]$ /software/cerebras/python3.8/bin/python3.8 -m venv venv_cerebras_pt
[jaiaid@cer-login-02 R_2.3.0]$ ls
venv_cerebras_pt
[jaiaid@cer-login-02 R_2.3.0]$ source venv_cerebras_pt/bin/activate
```

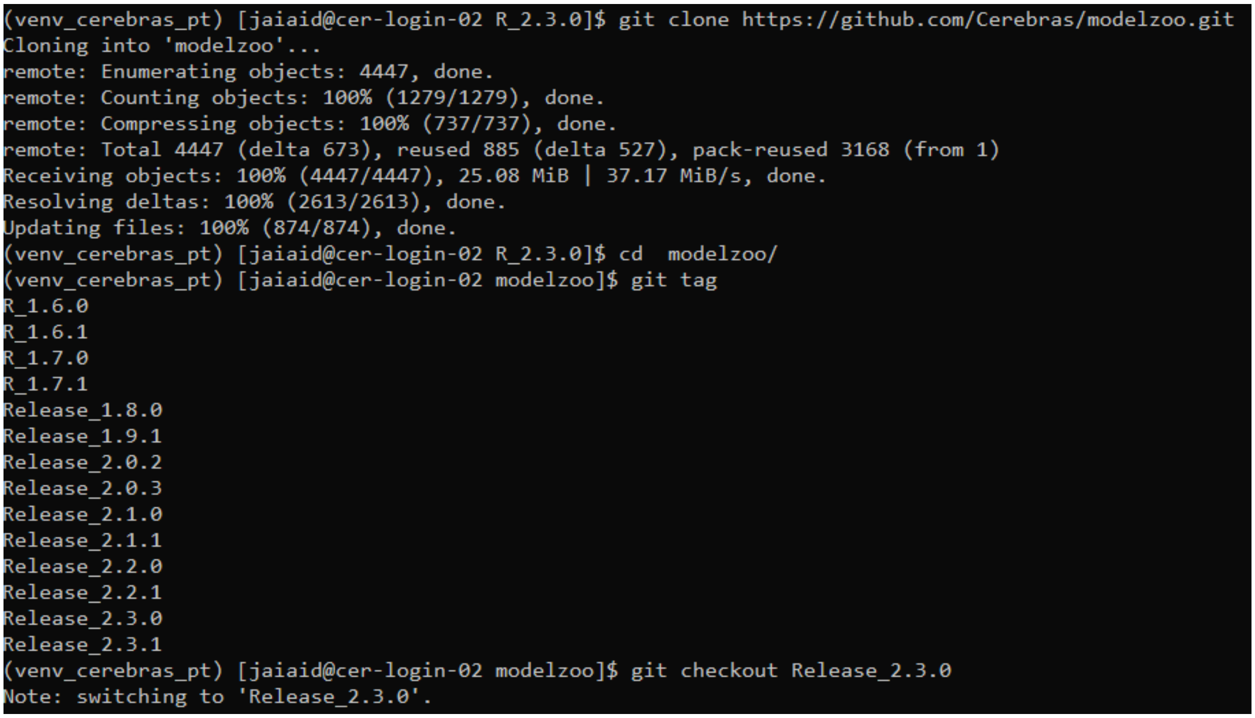


```
(venv_cerebras_pt) [jaiaid@cer-login-02 R_2.3.0]$ pip install --upgrade pip
Collecting pip
  Downloading pip-24.3.1-py3-none-any.whl (1.8 MB)
     |████████████████████████████████| 1.8 MB 14.0 MB/s
Installing collected packages: pip
  Attempting uninstall: pip
    Found existing installation: pip 20.2.3
    Uninstalling pip-20.2.3:
      Successfully uninstalled pip-20.2.3
Successfully installed pip-24.3.1
(venv_cerebras_pt) [jaiaid@cer-login-02 R_2.3.0]$ pip install cerebras_pytorch==2.3.0
Collecting cerebras_pytorch==2.3.0
```

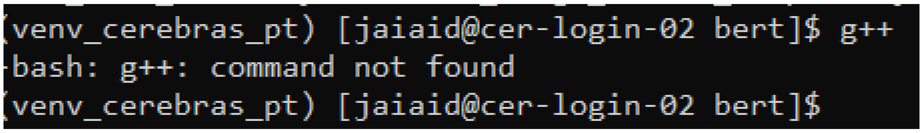


```
Successfully installed MarkupSafe-2.1.5 PyJWT-2.9.0 absl-py-2.1.0 cachetools-5.5.0 cerebras-appliance-2.
3.0 cerebras_pytorch-2.3.0 certifi-2024.8.30 charset-normalizer-3.4.0 dill-0.3.9 filelock-3.16.1 fsspec-
2024.10.0 google-auth-2.36.0 google-auth-oauthlib-0.4.6 grpcio-1.47.0 grpcio-tools-1.47.0 h5py-3.10.0 hd
f5plugin-5.0.0 idna-3.10 importlib-metadata-8.5.0 jinja2-3.1.4 markdown-3.7 mpmath-1.3.0 networkx-3.1 nu
mpy-1.24.4 nvidia-cublas-cu12-12.1.3.1 nvidia-cuda-cupti-cu12-12.1.105 nvidia-cuda-nvrtc-cu12-12.1.105 n
vidia-cuda-runtime-cu12-12.1.105 nvidia-cudnn-cu12-8.9.2.26 nvidia-cufft-cu12-11.0.2.54 nvidia-curand-cu
12-10.3.2.106 nvidia-cusolver-cu12-11.4.5.107 nvidia-cusparse-cu12-12.1.0.106 nvidia-nccl-cu12-2.20.5 nv
idia-nvjitlink-cu12-12.6.85 nvidia-nvtx-cu12-12.1.105 oauthlib-3.2.2 pandas-1.3.0 protobuf-3.15.6 psutil
-6.1.0 pyasn1-0.6.1 pyasn1-modules-0.4.1 python-dateutil-2.9.0.post0 pytz-2024.2 pyyaml-6.0.2 requests-2
.32.3 requests-oauthlib-2.0.0 rsa-4.9 six-1.16.0 sympy-1.13.3 tabulate-0.9.0 tblib-1.7.0 tensorboard-2.1
1.2 tensorboard-data-server-0.6.1 tensorboard-plugin-wit-1.8.1 torch-2.3.0 tqdm-4.67.1 triton-2.3.0 typi
ng-extensions-4.12.2 urllib3-1.26.20 werkzeug-3.0.6 wheel-0.45.1 zipp-3.20.2
```

3. Clone the repo and switch to version “Release_2.3.0”



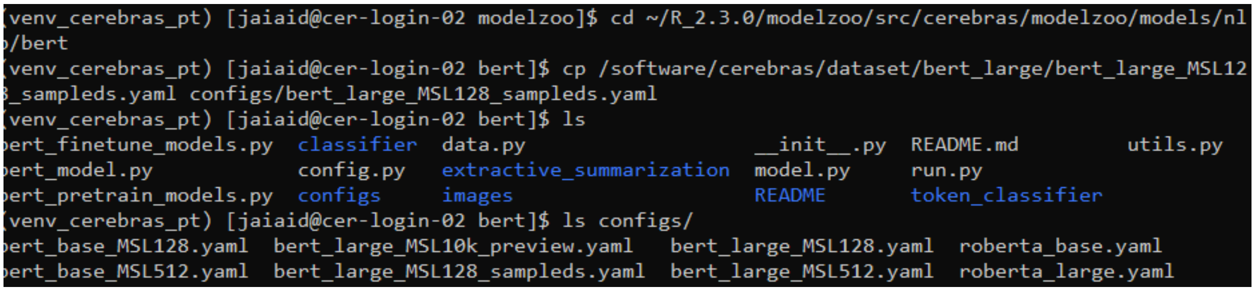
```
(venv_cerebras_pt) [jaiaid@cer-login-02 R_2.3.0]$ git clone https://github.com/Cerebras/modelzoo.git
Cloning into 'modelzoo'...
remote: Enumerating objects: 4447, done.
remote: Counting objects: 100% (1279/1279), done.
remote: Compressing objects: 100% (737/737), done.
remote: Total 4447 (delta 673), reused 885 (delta 527), pack-reused 3168 (from 1)
Receiving objects: 100% (4447/4447), 25.08 MiB | 37.17 MiB/s, done.
Resolving deltas: 100% (2613/2613), done.
Updating files: 100% (874/874), done.
(venv_cerebras_pt) [jaiaid@cer-login-02 R_2.3.0]$ cd  modelzoo/
(venv_cerebras_pt) [jaiaid@cer-login-02 modelzoo]$ git tag
R_1.6.0
R_1.6.1
R_1.7.0
R_1.7.1
Release_1.8.0
Release_1.9.1
Release_2.0.2
Release_2.0.3
Release_2.1.0
Release_2.1.1
Release_2.2.0
Release_2.2.1
Release_2.3.0
Release_2.3.1
(venv_cerebras_pt) [jaiaid@cer-login-02 modelzoo]$ git checkout Release_2.3.0
Note: switching to 'Release_2.3.0'.
```

4. Pip install the required package
   - Some wheels failed to be built due to “g++ not found”. Packages names are cymem, murmurhash
   - 

```
(venv_cerebras_pt) [jaiaid@cer-login-02 bert]$ g++
bash: g++: command not found
(venv_cerebras_pt) [jaiaid@cer-login-02 bert]$
```

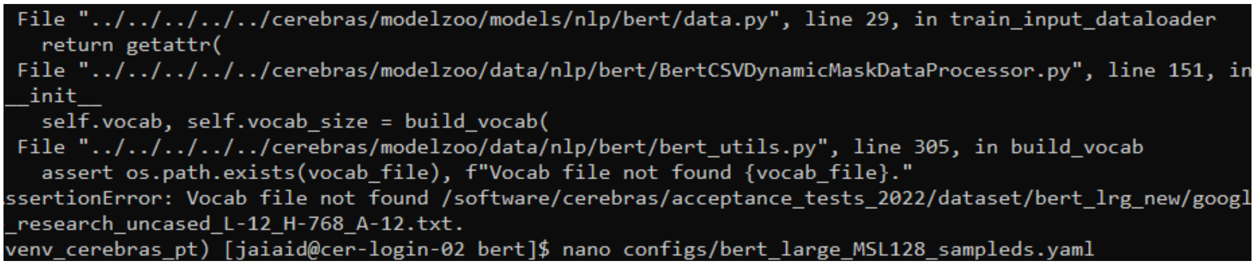
## Homework Workflow

1. Copy configuration files



```
(venv_cerebras_pt) [jaiaid@cer-login-02 modelzoo]$ cd ~/R_2.3.0/modelzoo/src/cerebras/modelzoo/models/nl
p/bert
(venv_cerebras_pt) [jaiaid@cer-login-02 bert]$ cp /software/cerebras/dataset/bert_large/bert_large_MSL12
8_sampleds.yaml configs/bert_large_MSL128_sampleds.yaml
(venv_cerebras_pt) [jaiaid@cer-login-02 bert]$ ls
bert_finetune_models.py  classifier  data.py                   __init__.py  README.md         utils.py
bert_model.py            config.py   extractive_summarization  model.py     run.py
bert_pretrain_models.py  configs     images                    README       token_classifier
(venv_cerebras_pt) [jaiaid@cer-login-02 bert]$ ls configs/
bert_base_MSL128.yaml  bert_large_MSL10k_preview.yaml  bert_large_MSL128.yaml  roberta_base.yaml
bert_base_MSL512.yaml  bert_large_MSL128_sampleds.yaml  bert_large_MSL512.yaml  roberta_large.yaml
```

2. Failed due to followings
   - jsonschema, torchvision==0.18, packaging, lm-eval not found
   - bigcode-eval installed from https://github.com/bigcode-project/bigcode-evaluation-harness
   - lm-eval required version is not clear so some imports were commented out
3. After solving above, finally failed due to dataset not found



```
 File "../../../../../cerebras/modelzoo/models/nlp/bert/data.py", line 29, in train_input_dataloader
   return getattr(
 File "../../../../../cerebras/modelzoo/data/nlp/bert/BertCSVDynamicMaskDataProcessor.py", line 151, in
__init__
   self.vocab, self.vocab_size = build_vocab(
 File "../../../../../cerebras/modelzoo/data/nlp/bert/bert_utils.py", line 305, in build_vocab
   assert os.path.exists(vocab_file), f"Vocab file not found {vocab_file}."
ssertionError: Vocab file not found /software/cerebras/acceptance_tests_2022/dataset/bert_lrg_new/googl
_research_uncased_L-12_H-768_A-12.txt.
venv_cerebras_pt) [jaiaid@cer-login-02 bert]$ nano configs/bert_large_MSL128_sampleds.yaml
```