SysDL Assignment 3 - AE17B020

Hyperparameter tuning on a GCP instance

Akash Haridas

Overview

- Secret gist with model definition code: https://gist.github.com/akasharidas/18b0fb7dc1d8449f8ea91162784c6aa4
- Total USD spent from credits: **$8.09**
- Best test set accuracy achieved: **42%**
- Total number of runs: **33**

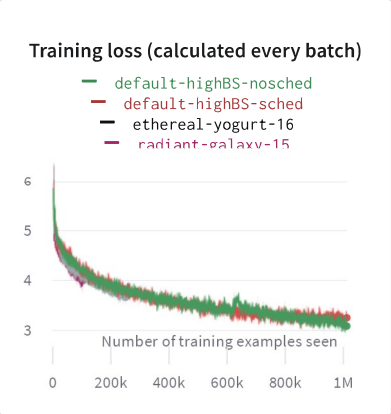
Part One:

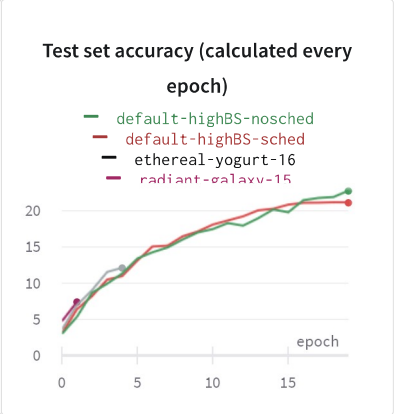
In the first phase, I implemented the model in PyTorch, wrote training code, tested and debugged to make sure it runs without errors and logs the metrics correctly. I also spent some time coming up with a workflow to run experiments. I found the following workflow to be effective:

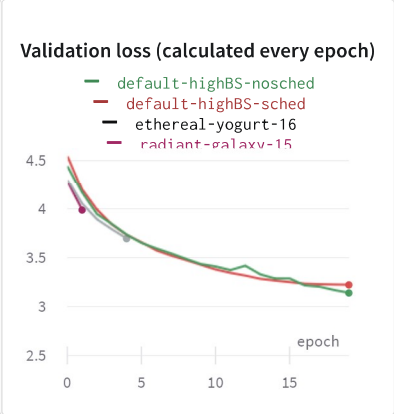
- Write/modify code on my local machine, and push the changes to a private Github repo.
- Pull from the repo on the GCP instance, and run the experiment.
- Monitor the experiment results on Weights and Biases, and go back to step 1 with appropriate changes.

Part Two

I did some **preliminary runs** to get a feel for which hyperparameters to tune and what ranges to try out in the subsequent systematic runs. Also in this phase I was learning how to use the GCP + Weights and Biases workflow effectively.





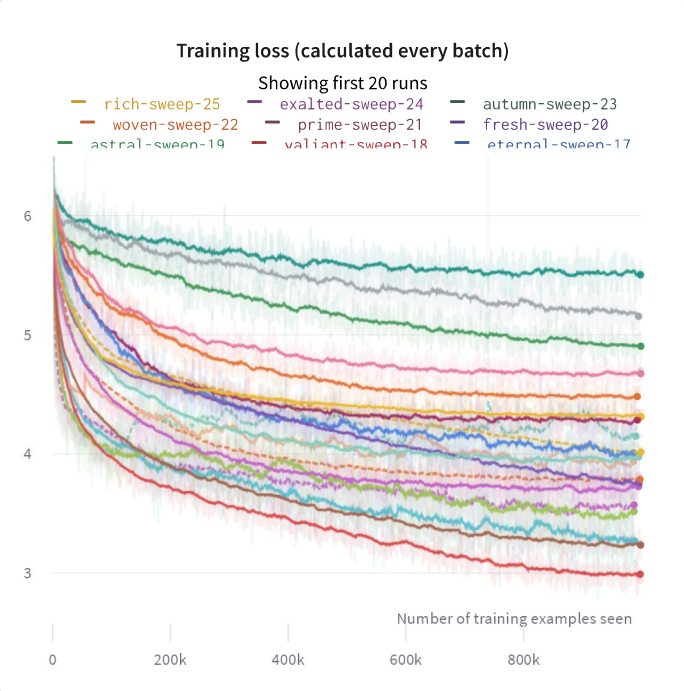


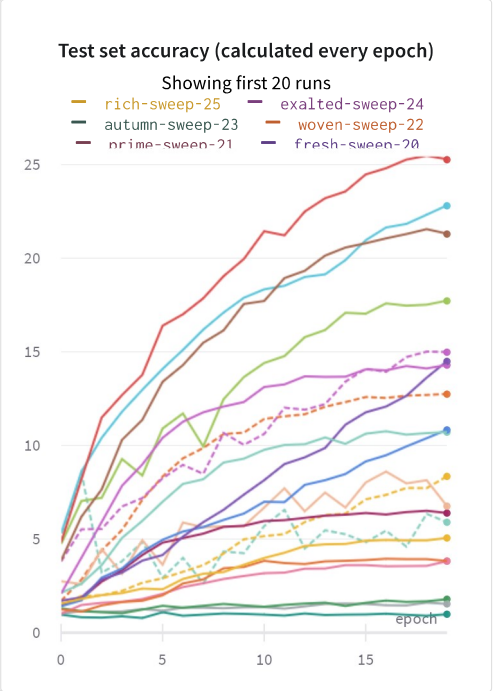
Part Three

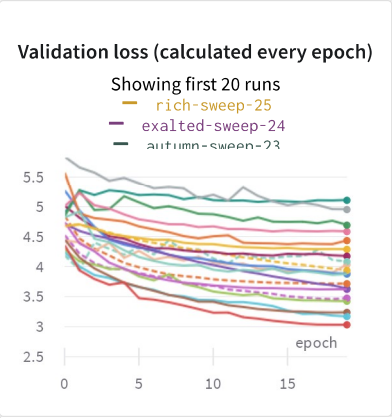
I ran a **coarse** hyperparameter sweep on Weights and Biases with the following settings:

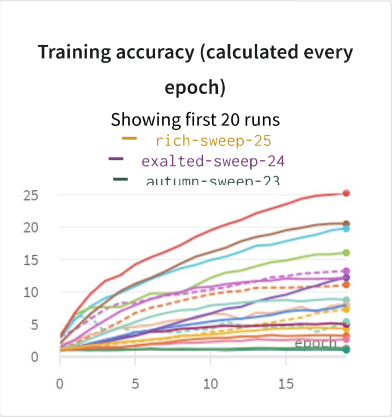
- Sampling method: Random sampling
- Batch size: [64, 128, 512]
- Learning rate: [0.0001, 0.0025, 0.0075, 0.001, 0.003]
- Momentum: [0, 0.1, 0.3, 0.6, 0.8, 0.9, 0.95]
- Optimizer: ['adam', 'sgd']
- Weight decay: [0, 1e-5, 3e-5, 1e-4, 3e-4, 1e-3]
- Scheduler: [True, False]

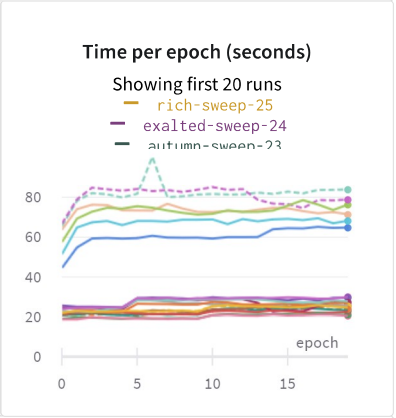
I did a total of **25 runs, each for 20 epochs.** I used the results of this sweep to identify roughly the best settings to use in the next part.

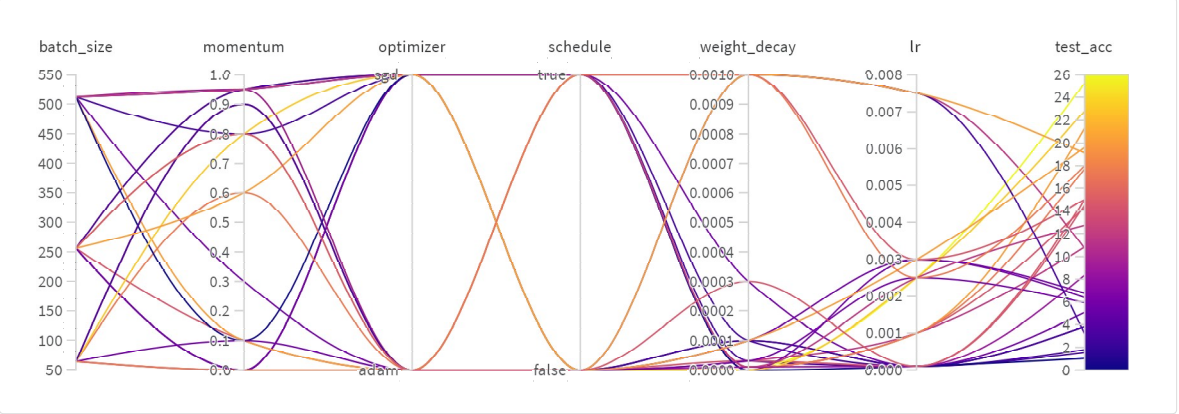












1. Batch size doesn't seem to have much impact on accuracy in the tested range of values, however a higher batch size makes the training significantly faster, so we shall use the batch size of 512 in the next section.
2. The best result was obtained with the Adam optimizer, but SGD came close.
3. It is **unclear** from this data whether using a learning rate schedule helps (*in hindsight: 20 epochs is too small for it to make a difference*)
4. A learning rate around 0.003 is best.

Run set 25

Name (25 visualized)	Runtime	batch_size	epochs	lr	momentum	optimizer
rich-sweep-25	8m 3s	512	20	0.0001	0.3	adam
exalted-sweep-24	9m 27s	256	20	0.001	0.95	sgd
autumn-sweep-23	9m 2s	256	20	0.0075	0	sgd
woven-sweep-22	8m 35s	256	20	0.0001	0.95	sgd
prime-sweep-21	6m 43s	512	20	0.0001	0.95	sgd
fresh-sweep-20	9m 25s	256	20	0.0001	0.1	adam
astral-sweep-19	6m 49s	512	20	0.0001	0.8	sgd
valiant-sweep-18	7m 29s	512	20	0.0025	0.95	adam
eternal-sweep-17	20m 4s	64	20	0.001	0	sgd
effortless-sweep-16	8m 48s	256	20	0.0001	0	sgd
crisp-sweep-15	10m 23s	256	20	0.003	0	sgd
ruby-sweep-14	7m 54s	512	20	0.001	0.1	adam
fresh-sweep-13	24m 20s	64	20	0.001	0.6	adam
good-sweep-12	24m 25s	64	20	0.003	0.9	adam
cerulean-sweep-11	7m 35s	512	20	0.0001	0.1	sgd
deep-sweep-10	22m 24s	64	20	0.0025	0.8	sgd
graceful-sweep-9	8m 5s	512	20	0.0001	0.95	adam
astral-sweep-8	26m 51s	64	20	0.003	0.9	adam
charmed-sweep-7	27m 17s	64	20	0.0025	0.1	adam
pious-sweep-6	7m 36s	512	20	0.0025	0.95	sgd

1-20 of 25

Part Four

In the final phase, I performed long runs (100+ epochs) with the best settings from the previous phase, and further manually tuned the settings to achieve my final result.

First I performed a run with the best settings from the previous part, but without a learning rate schedule (*ruby-senset-63*). I noticed that the loss became unsteady in the later part of training which meant that the LR was too high.

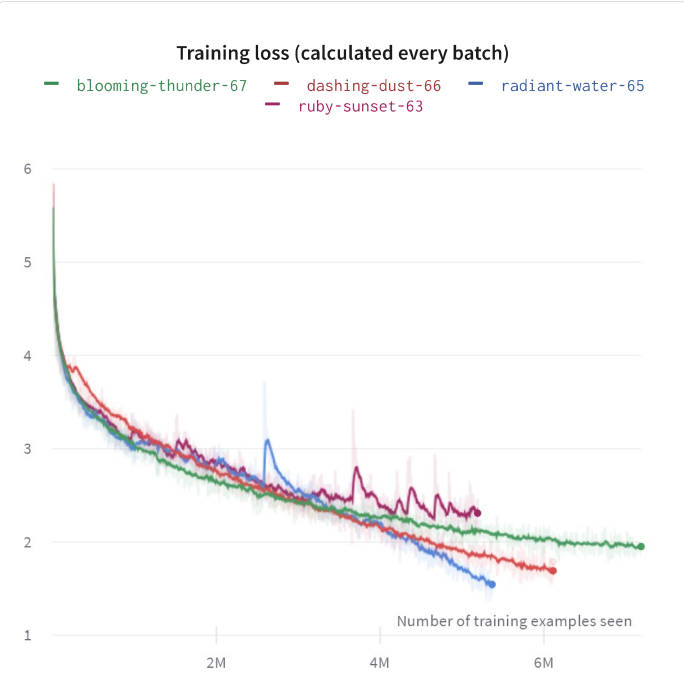
So I tried it again, but this time with Cosine Annealing LR schedule. This time the model started to

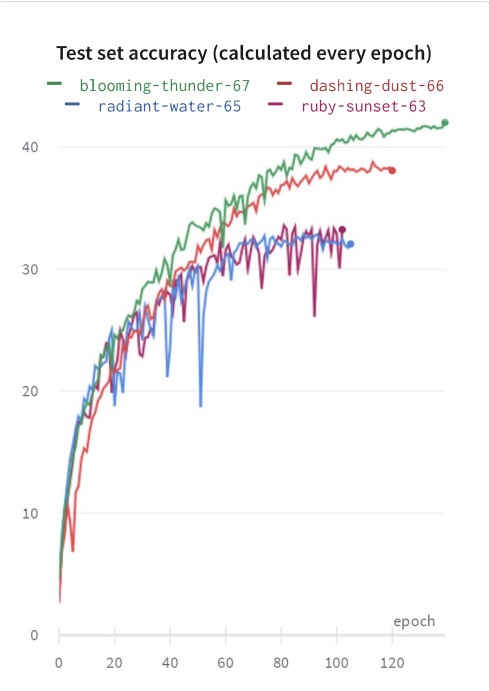
severely overfit after about 60 epochs (*radiant-water-65*).

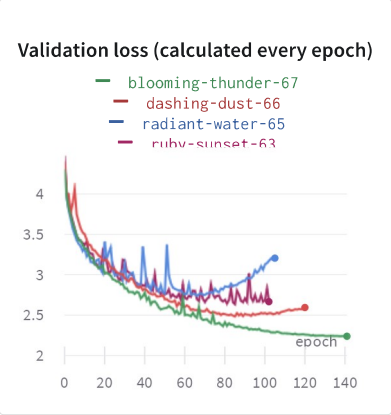
To curb the overfitting, I increased the **weight_decay** parameter of the optimizer which adds a penalty to the loss based on the magnitude of the weights. This helped a lot, but it was still overfitting after about 90 epochs (*dashing-dust-66*).

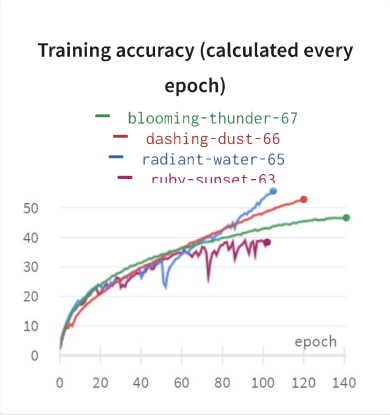
In my final attempt, I increased the weight_decay even more, and this time I achieved my best test set accuracy of **42%** after 140 epochs. My final settings are as follows:

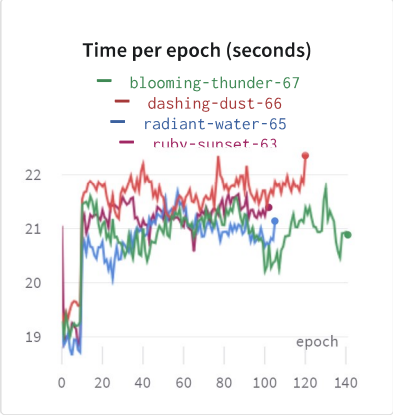
- Batch size - 512
- Adam optimizer
- LR - 0.0025
- Weight decay - 0.0003
- LR Schedule - Cosine Annealing











Created with ❤ on Weights & Biases.

https://wandb.ai/akasharidas/SysDL%20Assignment%203/reports/SysDL-Assignment-3-AE17B020--VmlldzoyODI1ODI