W&B and Ada

—

~ Hardik & Balu

AGENDA

- To introduce you to WandB and its key features, to explain the benefits of experiment tracking in machine learning, and to show you how to use WandB to improve your workflow

- Introduction- experiment tracking
- What is WandB?
- Let's see some code
- Overview of WandB sweep
- Overview, applications and functionality in WandB website

- Basics of Ada to give you the power of more compute

- Introduction to Ada
- Basic commands

- Doubts

Tracking experiments with WandB

Experiment tracking using WandB

- **Experiment tracking :** Experiment tracking refers to the process of systematically logging and organizing experiments in order to better understand the outcomes of different machine learning models and algorithms.

- **Importance of experiment tracking in machine learning** : Experiment tracking is critical for machine learning practitioners as it enables them to keep track of their work and reproduce their results. This is particularly important in complex projects where multiple people are working on different parts of the codebase.

Experiment tracking using WandB

There is no fun in analysing this

- {'epoch': 1, 'train_loss': 1.1452948740124702, 'Eval_loss':0.7713702845573426, 'train_ap_score':0.6731415009678289, 'eval_ap_score': 0.8116030823185824, 'lr': 0.0001}
- {'epoch': 2, 'Train_loss' : 0.5890168227255345, 'eval_loss':0.6724603283405304, 'train_ap_score':0.8755422621356335, 'eval_ap_score': 0.8489012873912365, 'lr': 0.0001}
- {'epoch': 3, 'Train_loss': 0.3640854485332966, 'eval_loss':0.6902201867103577, 'train_ap_score':0.9443318746126932, 'eval_ap_score': 0.8558312675174203, 'lr': 0.0001}
- {'epoch': 4, 'Train_loss': 0.20835940316319465, 'eval_loss':0.733428498506546, 'train_ap_score':0.9796843642719427, 'eval_ap_score': 0.8559268354265821, 'lr': 0.0001}
- {'epoch': 5, 'Train_loss': 0.1133101735264063, 'eval_loss':0.8106619369983673, 'train_ap_score':0.9937608794513503, 'eval_ap_score': 0.8549025780413813, 'lr': 0.0001}

Experiment tracking using WandB

Without visualization/tracking, it is tough to answer queries like-

- How long does it take to run your experiments?
- Around what epoch does it start to overfit?
- When scheduler updated the learning rate, how much did that affect the ther metrics? Did it even trigger?!
- How to compare X different runs that only have change in one parameter? Which one to choose?
- Many more…..

One place for all you experiments

- WandB is a powerful experiment tracking tool that helps machine learning practitioners to keep track of their models, datasets, and experiments. WandB offers a range of features including real-time visualization, hyperparameter tuning, and experiment comparison.
- Code → github | Exp. → WandB
- Easy to configure and use!

Lets jump to code

Log some configs and experiment data to WandB.
Head over to:

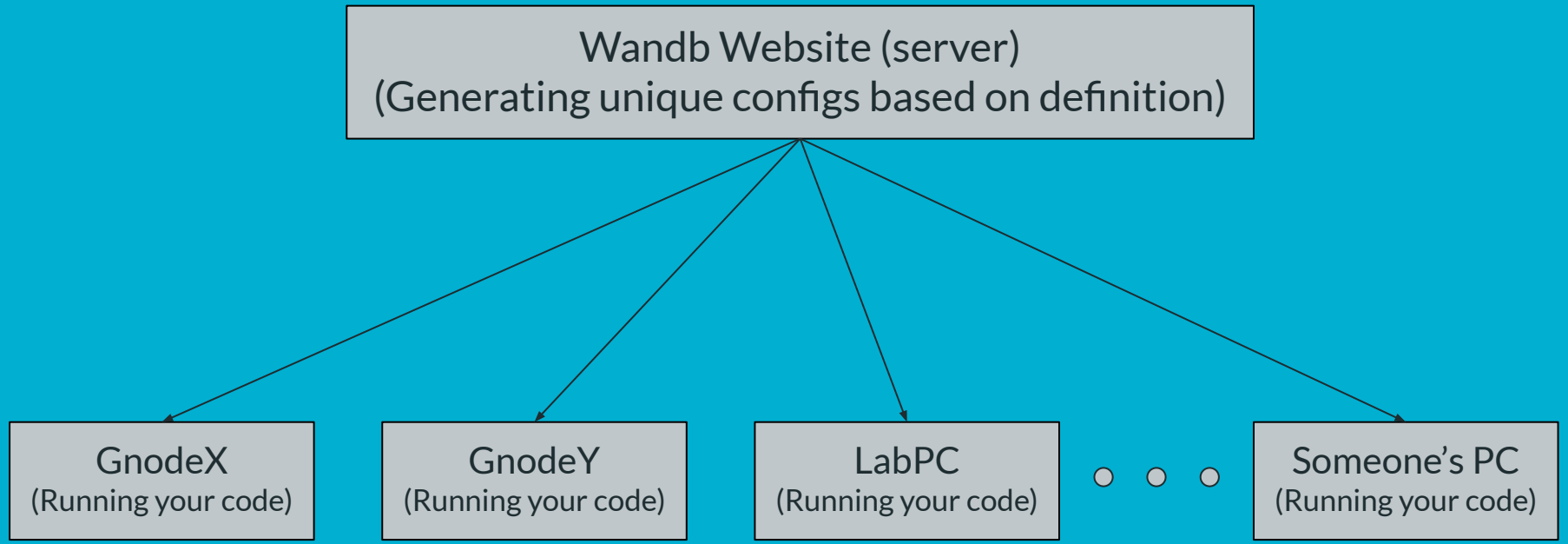
Log some configs and experiment data to WandB

Introduction to Weights & Biases :
https://colab.research.google.com/drive/1pKPxCRbLzwvhRvMmmCAYttpq53D0aW2M?usp=sharing

OR

https://tinyurl.com/3fpr8ne3

WandB Sweeps



Application of Sweeps

Generating configs automatically

- Efficient ablation management: Hassle free auto execution of every combination of varying parameters.
- Grid Search: Can estimate feature importance (can be selected manually) based on metric objective.
- Visualizing and Comparing multiple runs (of our choice) through parallel coordinates plot.
- We can engage as many gnodes (agents/workers) we want just by mentioning sweep-id while submitting batch job for each gnode. This parallely completes all the runs expected in that sweep. No need to submit usual batch job (with all pre-requirements of data and code satisfied for that gnode).
- So it does have a lot of perks, Agree? 🙇‍♂️ 💁‍♂️ 🙆‍♀️

How does a sweep look?

```python
import wandb

# Example sweep configuration
sweep_configuration = {
    "method": "random",
    "name": "sweep",
    "metric": {"goal": "maximize", "name": "val_acc"},
    "parameters": {
        "batch_size": {"values": [16, 32, 64]},
        "epochs": {"values": [5, 10, 15]},
        "lr": {"max": 0.1, "min": 0.0001},
    },
}

sweep_id = wandb.sweep(sweep=sweep_configuration, project="project-name")
```

Lets jump to code

WandB Sweeps :
https://colab.research.google.com/drive/1hn38J0BRTpGc-gt05kqTDYfokBdZuZ11?usp=sharing%5C#scrollTo=-i6wtdnVf5Dk

OR

https://tinyurl.com/2s4xkpm5

```
Every 2.0s: nvidia-smi                                                                  gnode031: Fri May 24 11:07:49 2024

Fri May 24 11:07:49 2024
+-----------------------------------------------------------------------------------------+
| NVIDIA-SMI 530.41.03              Driver Version: 530.41.03      CUDA Version: 12.1     |
|-----------------------------------------+----------------------+----------------------+
| GPU  Name                 Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf            Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|                                         |                      |               MIG M. |
|=========================================+======================+======================|
|   0  NVIDIA GeForce GTX 1080 Ti      Off| 00000000:02:00.0 Off |                  N/A |
| 23%   33C    P8               17W / 250W|      0MiB / 11264MiB |      0%      Default |
|                                         |                      |                  N/A |
+-----------------------------------------+----------------------+----------------------+
|   1  NVIDIA GeForce GTX 1080 Ti      Off| 00000000:03:00.0 Off |                  N/A |
| 23%   31C    P8                9W / 250W|      0MiB / 11264MiB |      0%      Default |
|                                         |                      |                  N/A |
+-----------------------------------------+----------------------+----------------------+
|   2  NVIDIA GeForce GTX 1080 Ti      Off| 00000000:82:00.0 Off |                  N/A |
| 23%   32C    P8                9W / 250W|      0MiB / 11264MiB |      0%      Default |
|                                         |                      |                  N/A |
+-----------------------------------------+----------------------+----------------------+
|   3  NVIDIA GeForce GTX 1080 Ti      Off| 00000000:83:00.0 Off |                  N/A |
| 23%   30C    P8                9W / 250W|      0MiB / 11264MiB |      0%      Default |
|                                         |                      |                  N/A |
+-----------------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------------------+
| Processes:                                                                              |
|  GPU   GI   CI        PID   Type   Process name                            GPU Memory |
|        ID   ID                                                             Usage      |
|=========================================================================================|
|  No running processes found                                                             |
+-----------------------------------------------------------------------------------------+
```

Ada : A necessary evil

```
[hardik.mittal@ada]~% sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
short        up    6:00:00      2   idle gnode[003,009]
long*        up   infinite      2 drain* gnode[034,077]
long*        up   infinite      4   resv gnode[043,045,067,084]
long*        up   infinite     38    mix gnode[001-002,010,012,021,023,026,029,040-041,
,085,087,092]
long*        up   infinite      5  alloc gnode[025,027,030,060,070]
long*        up   infinite     31   idle gnode[004-008,011,015-017,019-020,022,028,031-
,086,090-091]
ihub         up   infinite      1 drain* gnode111
ihub         up   infinite     10    mix gnode[097,099-101,103-104,107-110]
ihub         up   infinite      9   idle gnode[093-096,098,102,105-106,112]
plafnet2     up   infinite      2    mix gnode[115-116]
plafnet2     up   infinite      3   idle gnode[113-114,118]
rrc          up   infinite      1    mix gnode117
```

Who all have Ada accounts here?

Introduction

- Ada is an HPC (High-Performance Computing) cluster at IIIT Hyderabad. To connect to it, you need to be connected to IIIT intranet
- SLURM software is used as a job scheduler and resource manager in Ada.

- If you haven't received an Ada account, please fill this form. (available at the official hpc website)
- Official Documentation Website : https://hpc.iiit.ac.in/wiki/index.php/Ada_User_Guide

Ada configuration

```
(base) [hardik.mittal@ada ~]$ cat /etc/*release
CentOS Linux release 7.9.2009 (Core)
NAME="CentOS Linux"
VERSION="7 (Core)"
ID="centos"
ID_LIKE="rhel fedora"
VERSION_ID="7"
PRETTY_NAME="CentOS Linux 7 (Core)"
ANSI_COLOR="0;31"
CPE_NAME="cpe:/o:centos:centos:7"
HOME_URL="https://www.centos.org/"
BUG_REPORT_URL="https://bugs.centos.org/"
```

Ada Master Node
(containing share storage)

```
(base) hardik.mittal@gnode012: $ cat /etc/*release
DISTRIB_ID=Ubuntu
DISTRIB_RELEASE=18.04
DISTRIB_CODENAME=bionic
DISTRIB_DESCRIPTION="Ubuntu 18.04.6 LTS"
NAME="Ubuntu"
VERSION="18.04.6 LTS (Bionic Beaver)"
ID=ubuntu
ID_LIKE=debian
PRETTY_NAME="Ubuntu 18.04.6 LTS"
VERSION_ID="18.04"
HOME_URL="https://www.ubuntu.com/"
SUPPORT_URL="https://help.ubuntu.com/"
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/te
VERSION_CODENAME=bionic
UBUNTU_CODENAME=bionic
```

GnodeX
(Running your code)

GnodeY
(Running your code)

GnodeZ
(Running your code)

iHub, RRC, plafnet

*Mostly if the code doesn't run, it is an issue with your code but sometimes it is the node that is bad.

*Node 01-40 contains 4 GeForce GTX 1080 Ti GPUs each while nodes 43-92 contain 4 GeForce RTX 2080 Ti GPUs each.

Storage

```
Disk quotas for user hardik.mittal (uid 2876):
     Filesystem   space   quota   limit   grace   files   quota   limit   grace
        /share3    186G    200G    201G            200k       0       0
          /home  14815M  25600M  26624M            150k    300k    305k
```

Storage Directory	Access	Maximum Quota
/home2/$USER or ~	Main node and allocated nodes (home)	25 GB
/share1/$USER	Main node only (not on allocated nodes)	100 GB
/scratch/$USER	Allocated nodes only (not on main node)	2 TB (7 days) [1]

[1]: Files are cleared after a hard 7-day limit. Storage is collective, big, but temporary

misc

- Storing your SSH public key on Ada for passwordless login (ssh-keygen, ssh-copy-id -i ~/.ssh/mykey user@host)
- Logging to Ada through VScode
- Don't log in to Ada but a Gnode through VScode
  - Only 100 processes are allowed per account on the master node of Ada. Running VScode there will lead to exceeding that limit leading to crashing of the account.
- SSH config to directly log into a gnode

```
Host gnode*
  HostName %h
  ProxyJump hardik.mittal@ada
  User hardik.mittal
```

misc

- To see your Ada files locally (for Linux) : sftp://hardik.mittal@ada/
- Before starting to use, you need to install conda/miniconda/mamba..
  - Helps you keep different environments for different projects having different dependencies

VERY IMPORTANT

- `watch nvidia-smi`
- Do not run empty batch jobs to just keep a gnode with you.
  - If the admin finds it, your account will be suspended
- Always acquire the gnodes with the config of 9:1 for cpu:gpu
  - This helps other to acquire the gnode to copy their data if they have saved stuff in its scratch and continue working somewhere else
- If you find that someone has acquired all the resources of a gnode and you want to retrieve your data from that gnode, you can *finger* them
  - Find out the username of who is using that gnode from `squeue -w <gnode>`
  - Find out their email id from `finger <username>` and mail them to relinquish atleast one CPU.

Link to these slides

- https://docs.google.com/presentation/d/1ycP_7qe255JJ7ODp3iPiWvlkPVwGgYn1SWvGOcI_KKY/edit#slide=id.p

Other resources

- Research-Starter-Kit : https://github.com/dheerajpreddy/Research-Starter-Kit
- Ada cluster tutorial : https://docs.google.com/presentation/d/1d5otkiIrFH0xsyTO2Z3BqycL7V5LpbUNHpT7biUEcmk/edit#slide=id.p34
- kharyal/jupyter-notebook-on-servers : https://github.com/kharyal/jupyter-notebook-on-servers
- server cheatsheet : https://docs.google.com/document/d/1S-JHlJ4T-uHSECvXmjiGcYwit7RtMDb8a_Iu6VtoSCY/edit
- Youtube-tutorial : https://www.youtube.com/watch?v=U3_pPJgs2Fg
- Wiki page : https://hpc.iiit.ac.in/wiki/index.php/Ada_User_Guide
- Ada Notion Guide : https://avneesh-m.notion.site/ADA-Guide-35d79bbff0b5400db2b7bef3d8f239d2
- IIITH Community Guide : https://saishubodh.notion.site/IIITH-Research-Paper-Reading-Group-CV-Robot-DL-Research-IIITH-Community-2656246269d24ab4818b5da24020a3d5

Really nice courses

- To brush up math concepts for ML (link)
- Stanford CS231n (link)
- UMich EECS 498-007 (link)
- CS25 (link)
  - Reading Material of first three lectures
- UvA Deep Learning Lecture (link)
  - Need not do the whole course, just the ones which look important

- For pytorch : https://www.learnpytorch.io/