

**DSPRO2**

## **Experiment Tracking with W&B**

Pascal Wullschleger

Hochschule Luzern

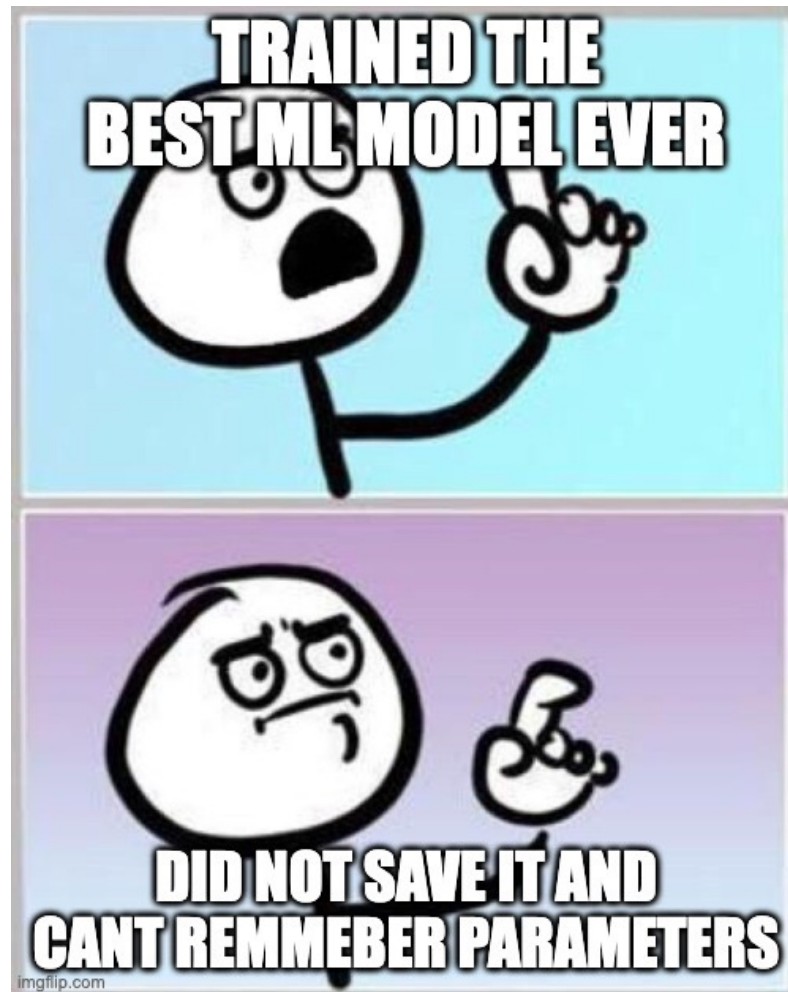
# Outline

1. Short input with overview and setup for W&B
2. Tutorial
3. Examples from our research projects
4. Individual set-up for your DSPRO2 projects

# What can we track?

- Git commit hash
- Configuration files
- Datasets used (depends on size)
- Model and training parameters
- Evaluation metrics
- Models
- Visualizations (learning curves, etc.)
- Example outputs
- Anything you like

## Why track experiments?



# **Why track experiments?**

- Reproducibility
- Transparency
- Comparing models & tuning parameters
- Efficient Workflow

## Tools

- Mlflow (open source)
- Weights & Biases (commercial, free for personal use)
- Tensorboard (open source, intended for tensorflow)
- Manual tracking (not recommended)
- Loads of other tools available



# **Weights & Biases**

Most tools offer more or less the same functionality

We will look at W&B, since

- Free for personal use
- Minimal set-up, no hosting
- More interactive than for example Mlflow & tensorboard
- Good integration with standard libraries

## **What can W&B do?**

- Experiment tracking
- Visualization of tracked stats
- Hyper parameter sweeps
- Managing various artifacts



## Set-up

1. [Sign up](#) with GitHub, Google MS or create a new account
2. Create API token at User Settings -> Danger Zone -> API keys
3. Install the PyPI package and log in

```
pip install wandb  
wandb login <api-key>
```

# Minimal Example

```
import wandb
import random

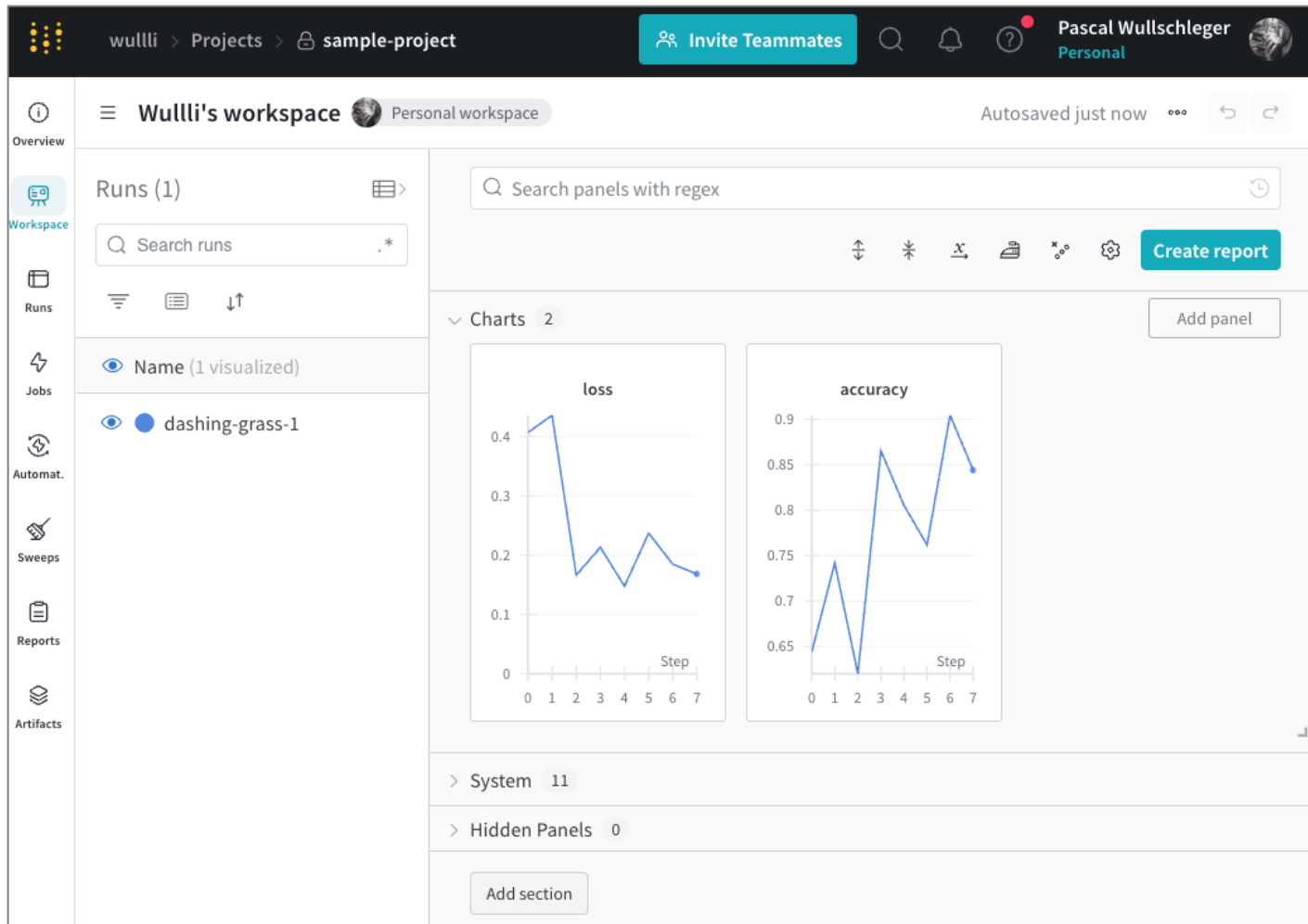
epochs, lr = 10, 0.01
run = wandb.init(
    project="sample-project", # Set the project where this run will be logged
    config={ # Track hyperparameters and run metadata
        "learning_rate": lr,
        "epochs": epochs
    })

offset = random.random() / 5

# simulating a training run
for epoch in range(2, epochs):
    acc = 1 - 2**(-epoch - random.random() / epoch - offset)
    loss = 2**(-epoch + random.random() / epoch + offset)
    print(f"epoch={epoch}, accuracy={acc}, loss={loss}")
    wandb.log({"accuracy": acc, "loss": loss})
```

<https://docs.wandb.ai/quickstart>

# Minimal Example



# **Reproducibility**

In order to reproduce results make sure you at least log the following

1. Version of your code-base and dataset
2. Model config & parameters
3. Random seeds

## **W&B Links**

Tutorials

<https://docs.wandb.ai/tutorials>

Developer guide

<https://docs.wandb.ai/guides/track/log/log-tables>