

# torchdistill Meets Hugging Face Libraries for Reproducible, Coding-Free Deep Learning Studies: A Case Study on NLP

Yoshitomo Matsubara\* (University of California, Irvine) \*This work was done prior to joining Amazon



```
$ pip3 install torchdistill
```

## torchdistill

- PyYAML configuration-driven ML OSS build on PyTorch
- Lower barriers to **reproducible, coding-free** deep learning / knowledge distillation studies

A typical script for model training

```
$ python3 run_task.py \  
  --model_name_or_path bert-large-uncased \  
  --dataset_name mnli \  
  --do_train \  
  --do_eval \  
  --max_seq_length 128 \  
  --per_device_train_batch_size 32 \  
  --per_device_eval_batch_size 32 \  
  --learning_rate 3e-5 \  
  --num_train_epochs 3 \  
  --optim adamw_torch \  
  --adam_epsilon 1e-8 \  
  --logging_dir ./my_experiment_20231206/ \  
  --output_dir ./my_experiment_20231206/
```

```
$ python3 run_task.py \  
  --config your_config.yaml \  
  --run_log your_log.txt
```

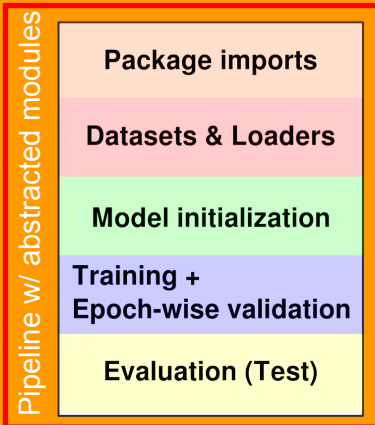
ML Tasks with  
Packages of Your Choice e.g.,  
transformers datasets  
evaluate accelerate timm



Almost everything to  
design an experiment



Abstracted modules



- Dependencies
- Datasets
- Preprocessing
- Data loaders
- Models
- Model wrappers
- Forward hooks
- Forward inferences
- Loss functions
- Optimizer
- LR scheduler
- Stage-wise config and more!

Evaluation result, **training log**,  
model weights