

# Training

For the training of RepNeXt, we refer to the training code of [ConvNeXt](#). You can download the four files `datasets.py`, `engine.py`, `utils.py` and `main.py` into your own folder and organize your data into the format of Imagefolder as follows.

```
basedir
├─ data
│   ├── train
│   │   ├── class_1
│   │   ├── class_2
│   │   └─ class_n
│   └─ val
│       ├── class_1
│       ├── class_2
│       └─ class_n
├─ datasets.py
├─ engine.py
├─ main.py
├─ reparameterizer.py
├─ repnext.py
└─ utils.py
```

To ensure the smooth running of the program, please delete the LayerScale part in `main.py` and import `repnext` in `main.py`

The common training schemes we use are as follows.

training schemes	configuration
Optimizer	AdamW
Learning rate scheduler	Cosine
Warmup	20-epoch
Auto augment	Trivial Augment (rand-m9-mstd0.5-inc1)
Color jitter	0.4
Label smoothing	0.1
Random erasing	0.25
Mixup	0.8
CutMix	1.0
FixRes	224 x 224 for training, 256 x 256 for validation
EMA	use model ema

training schemes	configuration
Weight decay	0.05
Stochastic depth	0.1 ~ 0.3

The above settings for training schemes are relatively mature, you can directly use these settings to train on your own dataset, take training ImageNet-1K as an example, the corresponding training commands are shown below.

## Train repnext\_u3\_tiny on ImageNet-1K

```
python -m torch.distributed.launch --nproc_per_node=8 main.py --model repnext_u3_tiny --epochs 450 --weight_decay 0.05 --color_jitter 0.4 --aa rand-m9-mstd0.5-inc1 --smoothing 0.1 --reprob 0.25 --mixup 0.8 --cutmix 1.0 --use_amp true --drop_path 0.1 --batch_size 512 --lr 4e-3 --update_freq 1 --model_ema true --model_ema_eval true --data_set image_folder --data_path /path/to/imagenet-1k/train --eval_data_path /path/to/imagenet-1k/val --output_dir /path/to/save_results
```

## Train repnext\_u3\_small on ImageNet-1K

```
python -m torch.distributed.launch --nproc_per_node=8 main.py --model repnext_u3_small --epochs 450 --weight_decay 0.05 --color_jitter 0.4 --aa rand-m9-mstd0.5-inc1 --smoothing 0.1 --reprob 0.25 --mixup 0.8 --cutmix 1.0 --use_amp true --drop_path 0.2 --batch_size 256 --lr 4e-3 --update_freq 2 --model_ema true --model_ema_eval true --data_set image_folder --data_path /path/to/imagenet-1k/train --eval_data_path /path/to/imagenet-1k/val --output_dir /path/to/save_results
```

## Train repnext\_u3\_base on ImageNet-1K

```
python -m torch.distributed.launch --nproc_per_node=8 main.py --model repnext_u3_base --epochs 450 --weight_decay 0.05 --color_jitter 0.4 --aa rand-m9-mstd0.5-inc1 --smoothing 0.1 --reprob 0.25 --mixup 0.8 --cutmix 1.0 --use_amp true --drop_path 0.3 --batch_size 256 --lr 4e-3 --update_freq 2 --model_ema true --model_ema_eval true --data_set image_folder --data_path /path/to/imagenet-1k/train --eval_data_path /path/to/imagenet-1k/val --output_dir /path/to/save_results
```