

Sumário dos erro

[Sumário dos erro](#)

[Erro #0](#)

[Erro #1: AttributeError: type object 'Trainer' has no attribute 'add_argparse_args'](#)

[Erro #2](#)

[Erro #3 biblioteca taming](#)

[Erro #4](#)

[Erro #5](#)

[Erro #6](#)

[Erro #7](#)

[Erro #8](#)

[Erro #9: dataloader_idx](#)

[Erro #10: 'root_gpu'](#)

[Erro #11: on_train_batch_start](#)

[Erro #12: on_train_epoch_end](#)

Erro #1: AttributeError: type object 'Trainer' has no attribute 'add_argparse_args'

Mensagem de erro:

File "main.py", line 469, in <module>

 parser = Trainer.add_argparse_args(parser)

AttributeError: type object 'Trainer' has no attribute 'add_argparse_args'

Observações:

<https://github.com/Lightning-AI/pytorch-lightning/pull/16708> Aparentemente tanto o 'add_argparse_args' como o parse_argparse_args foram removidos do Lightning.

Tentativa 1:

A função get_parser() retorna os argumentos, mas como add_argparse_args não está mais disponível, eu tentei passar todos eles manualmente, mas ao executar eu recebi o erro

TypeError: __init__() got an unexpected keyword argument para cada um deles.

```
parser = Trainer(  
    name = parser.name,  
    resume = parser.resume,  
    base = parser.base,  
    train = parser.train,  
    project = parser.project,  
    debug = parser.debug,  
    seed = parser.seed,  
    postfix = parser.postfix,  
    logdir = parser.logdir,
```

```
scale_lr = parser.scale_lr,  
gpus = parser.gpus,  
accelerator = parser.accelerator,  
)
```

Tentativa 2: Então tentei passar com `parser = Trainer(**parser.__dict__)`, mas também recebi o erro `TypeError: __init__() got an unexpected keyword argument`.

Tentativa 3: Então tentei passar sem os argumentos com `parser = Trainer()`, com isso essa parte não apresentava mais erro, mas o erro vinha na parte seguinte, onde:

`opt, unknown = parser.parse_known_args()` apresentava o erro:

`AttributeError: 'Trainer' object has no attribute 'parse_known_args'`

Tentativa 4: Eu mantive apenas o `parser = get_parser()` e removi o `parser = parser.parse_args()` e o `parser = Trainer()`. não sei ainda se isso vai ser necessário no futuro.

Solução: A Tentativa 4 ficou como solução final.

Erro #2

Na operação `nondefault_trainer_args(opt)`, eu estava tendo o mesmo problema que o Erro #1, com o trecho `parser = Trainer.add_argparse_args(parser)`. Para resolver eu removi esse trecho como no Erro #1.

Erro #3 biblioteca taming

Faltando a biblioteca `taming`, no caso a única que encontrei foi a `taming-transformers`.

```
from ldm.models.autoencoder import VQModelInterface, IdentityFirstStage, AutoencoderKL  
File "/trabalho/latent-diffusion/ldm/models/autoencoder.py", line 6, in <module>  
    from taming.modules.vqvae.quantize import VectorQuantizer2 as VectorQuantizer  
ModuleNotFoundError: No module named 'taming'
```

Tentativa 1: instalar o `taming-transformers` com `pip install taming-transformers`

Tentativa 2(funcionou): instalar o `taming-transformers` com `pip install taming-transformers-rom1504`

Erro #4

Traceback (most recent call last):

```
File "main.py", line 599, in <module>
    trainer_kwargs["logger"] = instantiate_from_config(logger_cfg)
File "/trabalho/latent-diffusion/ldm/util.py", line 85, in instantiate_from_config
    return get_obj_from_str(config["target"])(**config.get("params", dict()))
File "/trabalho/latent-diffusion/ldm/util.py", line 93, in get_obj_from_str
    return getattr(importlib.import_module(module, package=None), cls)
AttributeError: module 'pytorch_lightning.loggers' has no attribute 'TestTubeLogger'
```

Solução: Como o 'TestTubeLogger' não está mais disponível no 'pytorch_lightning 2.2.1 eu troquei pela alternativa disponível no próprio código.

Alterando o `default_logger_cfg = default_logger_cfgs["testtube"]` para
`default_logger_cfg = default_logger_cfgs["wandb"]`

também foi necessário instalar a biblioteca wandb com `pip install -U 'wandb>=0.12.10'`

Erro #5

```
File "/trabalho/latent-diffusion/main.py", line 315, in __init__
    pl.loggers.TestTubeLogger: self._testtube,
AttributeError: module 'pytorch_lightning.loggers' has no attribute 'TestTubeLogger'
```

Na classe `class ImageLogger` foi preciso alterar o `testtube` para o `WandbLogger` com as seguintes modificações:

```
def _wandb(self, pl_module, images, batch_idx, split):
    for k, image in images.items():
        try:
            pl_module.logger.experiment.log({f"{split}/{k}": [wandb.Image(image)]},
            step=pl_module.global_step)
        except wandb.errors.Error as e:
            print(f"Error logging image to Wandb: {e}")
```

Atualize o dicionário ``self.logger_log_images``:

```
self.logger_log_images = {
    'WandbLogger': self._wandb,
}
```

Erro #6

Traceback (most recent call last):

```
File "main.py", line 706, in <module>
    trainer = Trainer.from_argparse_args(trainer_opt, **trainer_kwargs)
AttributeError: type object 'Trainer' has no attribute 'from_argparse_args'
```

Tentativa 1: troquei o `trainer = Trainer.from_argparse_args(trainer_opt, **trainer_kwargs)` por `trainer = Trainer(trainer_opt, **trainer_kwargs)`

Erro #7

Traceback (most recent call last):

```
File "main.py", line 707, in <module>
    trainer = Trainer(trainer_opt, **trainer_kwargs)
File
"/opt/conda/envs/ldm/lib/python3.8/site-packages/pytorch_lightning/utilities/argparse.py", line
70, in insert_env_defaults
    return fn(self, **kwargs)
File "/opt/conda/envs/ldm/lib/python3.8/site-packages/pytorch_lightning/trainer/trainer.py",
line 401, in __init__
    self._accelerator_connector = _AcceleratorConnector(
File
"/opt/conda/envs/ldm/lib/python3.8/site-packages/pytorch_lightning/trainer/connectors/acceler
ator_connector.py", line 134, in __init__
    self._check_config_and_set_final_flags(
File
"/opt/conda/envs/ldm/lib/python3.8/site-packages/pytorch_lightning/trainer/connectors/acceler
ator_connector.py", line 215, in _check_config_and_set_final_flags
    raise ValueError(
ValueError: You selected an invalid accelerator name:
`accelerator=Namespace(benchmark=True)`. Available names are: auto, cpu, cuda, mps,
tpu.
```

Tentativa 1: No trecho `trainer_config["accelerator"] = "ddp"`, foi necessário adicionar `trainer_config["gpus"] = [0]` para que ele realmente usasse a gpu e o trecho `trainer_opt = argparse.Namespace(**trainer_config)` foi removido e no lugar de `trainer_opt` deve ser usado o `trainer_config`.

Erro #8

```
logvar_t = self.logvar[t.to(self.device)].to(self.device)
RuntimeError: indices should be either on cpu or on the same device as the indexed tensor (cpu)
```

Solução: Torquei o `logvar_t = self.logvar[t.to(self.device)].to(self.device)` por

```
log_var_local = self.logvar.to(self.device)
logvar_t = log_var_local[t]
```

Erro #9: `dataloader_idx`

```
TypeError: on_validation_batch_end() missing 1 required positional argument:
'dataloader_idx'
```

Solução: Removi o parametro `dataloader_idx` da função **`on_train_batch_end`**

Erro #10: `'root_gpu'`

```
torch.cuda.reset_peak_memory_stats(trainer.root_gpu)
AttributeError: 'Trainer' object has no attribute 'root_gpu'
```

O root_gpu foi removido <https://github.com/Lightning-AI/pytorch-lightning/issues/11994>

Solução:

Na Função on_train_epoch_start na classe CUDACallback tive de alterar os trechos onde o trainer.root_gpu era usado por trainer.strategy.root_device.

Erro #11: on_train_batch_start

TypeError: on_train_batch_start() missing 1 required positional argument: 'dataloader_idx'

Problema: Em versões mais recentes o on_train_batch_start não usa mais o parâmetro dataloader_idx.

https://lightning.ai/docs/pytorch/stable/common/lightning_module.html

Solução:

No arquivo ldm/models/diffusion/ddp.py, tive de remover o parâmetro dataloader_idx da função on_train_batch_start.

Erro #12: on_train_epoch_end

Nessa função foram encontrados dois erros:

Error 1: TypeError: on_train_epoch_end() missing 1 required positional argument: 'outputs'

Error 2: AttributeError: 'Trainer' object has no attribute 'root_gpu'

Solução:

Para o erro 1 a solução foi remover o parâmetro 'outputs' da função, pois ele não estava sendo usado em nenhum ponto da função e estava causando o erro.

Para o erro 2 a solução foi a mesma usada no Erro #10, substituir o trainer.root_gpu era por trainer.strategy.root_device