

```


from datasets import load_dataset
import torch
from transformers import AutoTokenizer, AutoModelForSeq2SeqLM
from transformers import DataCollatorForSeq2Seq
from transformers import TrainingArguments, Trainer
from transformers import pipeline

```

```

model_name = "facebook/mbart-large-50"
tokenizer = AutoTokenizer.from_pretrained(model_name)
model = AutoModelForSeq2SeqLM.from_pretrained(model_name).to("cuda")

```

 /usr/local/lib/python3.12/dist-packages/huggingface_hub/utils/_auth.py:94: UserWarning:
The secret `HF_TOKEN` does not exist in your Colab secrets.
To authenticate with the Hugging Face Hub, create a token in your settings tab (<https://huggingface.co/settings/tokens>)
You will be able to reuse this secret in all of your notebooks.
Please note that authentication is recommended but still optional to access public model

```

tokenizer_config.json: 100% 531/531 [00:00<00:00, 22.0kB/s]

config.json: 1.42k/? [00:00<00:00, 29.7kB/s]

sentencepiece.bpe.model: 100% 5.07M/5.07M [00:04<00:00, 1.25MB/s]

special_tokens_map.json: 100% 649/649 [00:00<00:00, 16.8kB/s]


pytorch_model.bin: 100% 2.44G/2.44G [00:47<00:00, 67.3MB/s]

model.safetensors: 100% 2.44G/2.44G [00:33<00:00, 68.7MB/s]

generation_config.json: 100% 261/261 [00:00<00:00, 2.58kB/s]

```

```
dataset = load_dataset("ai4bharat/samanantar", "mr", split="train[:2%]")
```

 README.md: 11.4k/? [00:00<00:00, 1.03MB/s]

```

train-00000-of- 244M/244M [00:14<00:00, 19.3MB/s]
00002.parquet: 100%

train-00001-of- 244M/244M [00:12<00:00, 21.4MB/s]
00002.parquet: 100%

Generating train split: 100% 3627480/3627480 [00:10<00:00, 392334.19 examples/s]

```

```

tokenizer.src_lang = "en_XX"
tokenizer.tgt_lang = "mr_IN"

```

```

def preprocess(examples):
    inputs = examples['src']

```

```

targets = examples['tgt']
model_inputs = tokenizer(inputs, max_length=128, truncation=True, padding='max_length')
with tokenizer.as_target_tokenizer():
    labels = tokenizer(targets, max_length=128, truncation=True, padding='max_length')
model_inputs["labels"] = labels["input_ids"]
return model_inputs

```

```
tokenized = dataset.map(preprocess, batched=True)
```



Map: 100%

72550/72550 [00:22<00:00, 4154.46 examples/s]

```

/usr/local/lib/python3.12/dist-packages/transformers/tokenization_utils_base.py:4006: UserWarning:

```

```
seq2seq_collator = DataCollatorForSeq2Seq(tokenizer, model=model, padding='longest', return_tensors='pt')
```

```

args = TrainingArguments(
    output_dir="./mbart-en-mr",
    learning_rate=3e-5,
    per_device_train_batch_size=8,
    per_device_eval_batch_size=8,
    weight_decay=0.01,
    save_total_limit=2,
    num_train_epochs=1,
    fp16=True
)

```

```

trainer = Trainer(
    model=model,
    args=args,
    train_dataset=tokenized,
    tokenizer=tokenizer,
    data_collator=seq2seq_collator
)

```



```

/tmp/ipython-input-4122455342.py:1: FutureWarning: `tokenizer` is deprecated and will be removed in a future version.
trainer = Trainer(


```

```
trainer.train()
```

```

/usr/local/lib/python3.12/dist-packages/notebook/notebookapp.py:191: SyntaxWarning: in
| | | | '_' \/_` / _` | _/ -_)
wandb: Logging into wandb.ai. (Learn how to deploy a W&B server locally: https://wandb
wandb: You can find your API key in your browser here: https://wandb.ai/authorize?ref=
wandb: Paste an API key from your profile and hit enter: .....
wandb: WARNING If you're specifying your api key in code, ensure this code is not shar
wandb: WARNING Consider setting the WANDB_API_KEY environment variable, or running `wa
wandb: No netrc file found, creating one.
wandb: Appending key for api.wandb.ai to your netrc file: /root/.netrc
wandb: Currently logged in as: khetres3636 (khetres3636-na) to https://api.wandb.ai. U
Tracking run with wandb version 0.21.1
Run data is saved locally in /content/wandb/run-20250822_041419-n3b4lqrc
Syncing run fanciful-pyramid-5 to Weights & Biases (docs)
View project at https://wandb.ai/khetres3636-na/huggingface
View run at https://wandb.ai/khetres3636-na/huggingface/runs/n3b4lqrc

```

 [9069/9069 2:44:17, Epoch 1/1]

Step Training Loss

500	1.604600
1000	0.487000
1500	0.455500
2000	0.441600
2500	0.437900
3000	0.415300
3500	0.412600
4000	0.408200
4500	0.404200
5000	0.398700
5500	0.395900
6000	0.393600
6500	0.384800
7000	0.375600
7500	0.378100
8000	0.379100
8500	0.386400
9000	0.369600

```

/usr/local/lib/python3.12/dist-packages/transformers/modeling_utils.py:3917: UserWarni
warnings.warn(
TrainOutput(global_step=9069, training_loss=0.47308708049586146, metrics=
{'train_runtime': 9908.0587, 'train_samples_per_second': 7.322,
'train_steps_per_second': 0.915, 'total_flos': 1.96531580829696e+16, 'train_loss':

```

```
0.47308708049586146, 'epoch': 1.0})
```

```
model.save_pretrained("(en-mar)_model")
tokenizer.save_pretrained("(en-mar)_tokenizer")

⇒ ('(en-mar)_tokenizer/tokenizer_config.json',
   '(en-mar)_tokenizer/special_tokens_map.json',
   '(en-mar)_tokenizer/sentencepiece.bpe.model',
   '(en-mar)_tokenizer/added_tokens.json',
   '(en-mar)_tokenizer/tokenizer.json')
```