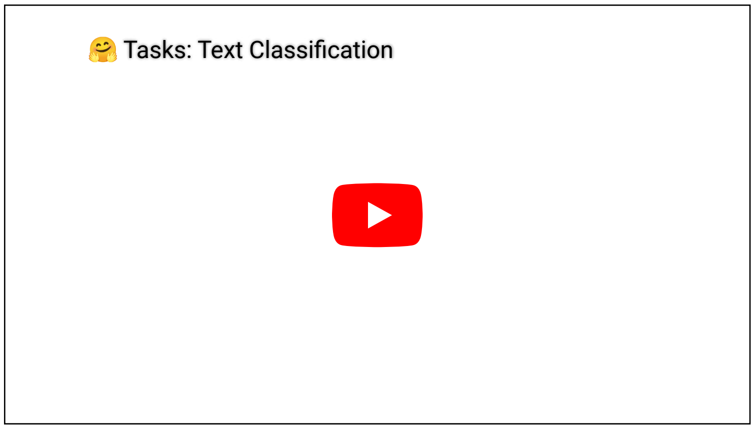
```
# Transformers installation
! pip install transformers datasets
# To install from source instead of the last release, comment t|he command above and uncomment the following one.
# ! pip install git+https://github.com/huggingface/transformers.git
```

```
Requirement already satisfied: transformers in /usr/local/lib/python3.10/dist-packages (4.40.1)
Requirement already satisfied: datasets in /usr/local/lib/python3.10/dist-packages (2.19.0)
Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-packages (from transformers) (3.13.4)
Requirement already satisfied: huggingface-hub<1.0,>=0.19.3 in /usr/local/lib/python3.10/dist-packages (from transformers) (
Requirement already satisfied: numpy>=1.17 in /usr/local/lib/python3.10/dist-packages (from transformers) (1.25.2)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.10/dist-packages (from transformers) (24.0)
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.10/dist-packages (from transformers) (6.0.1)
Requirement already satisfied: regex!=2019.12.17 in /usr/local/lib/python3.10/dist-packages (from transformers) (2023.12.25)
Requirement already satisfied: requests in /usr/local/lib/python3.10/dist-packages (from transformers) (2.31.0)
Requirement already satisfied: tokenizers<0.20,>=0.19 in /usr/local/lib/python3.10/dist-packages (from transformers) (0.19.1
Requirement already satisfied: safetensors>=0.4.1 in /usr/local/lib/python3.10/dist-packages (from transformers) (0.4.3)
Requirement already satisfied: tqdm>=4.27 in /usr/local/lib/python3.10/dist-packages (from transformers) (4.66.2)
Requirement already satisfied: pyarrow>=12.0.0 in /usr/local/lib/python3.10/dist-packages (from datasets) (14.0.2)
Requirement already satisfied: pyarrow-hotfix in /usr/local/lib/python3.10/dist-packages (from datasets) (0.6)
Requirement already satisfied: dill<0.3.9,>=0.3.0 in /usr/local/lib/python3.10/dist-packages (from datasets) (0.3.8)
Requirement already satisfied: pandas in /usr/local/lib/python3.10/dist-packages (from datasets) (2.0.3)
Requirement already satisfied: xxhash in /usr/local/lib/python3.10/dist-packages (from datasets) (3.4.1)
Requirement already satisfied: multiprocess in /usr/local/lib/python3.10/dist-packages (from datasets) (0.70.16)
Requirement already satisfied: fsspec[http]<=2024.3.1,>=2023.1.0 in /usr/local/lib/python3.10/dist-packages (from datasets)
Requirement already satisfied: aiohttp in /usr/local/lib/python3.10/dist-packages (from datasets) (3.9.5)
Requirement already satisfied: aiosignal>=1.1.2 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (1.3.1)
Requirement already satisfied: attrs>=17.3.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (23.2.0)
Requirement already satisfied: frozenlist>=1.1.1 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (1.4.1)
Requirement already satisfied: multidict<7.0,>=4.5 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (6.0.
Requirement already satisfied: yarl<2.0,>=1.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (1.9.4)
Requirement already satisfied: async-timeout<5.0,>=4.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (
Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/lib/python3.10/dist-packages (from huggingface-hub<1
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests->transform
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests->transformers) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests->transformers) (
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests->transformers) (
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.10/dist-packages (from pandas->datasets) (2.
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-packages (from pandas->datasets) (2023.4)
Requirement already satisfied: tzdata>=2022.1 in /usr/local/lib/python3.10/dist-packages (from pandas->datasets) (2024.1)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.8.2->pandas->dat
```

Text classification

Show code



Text classification is a common NLP task that assigns a label or class to text. Some of the largest companies run text classification in production for a wide range of practical applications. One of the most popular forms of text classification is sentiment analysis, which assigns a label like 🙂 positive, 🙁 negative, or 😐 neutral to a sequence of text.

This guide will show you how to:

1. Finetune DistilBERT on the IMDb dataset to determine whether a movie review is positive or negative.
2. Use your finetuned model for inference.

The task illustrated in this tutorial is supported by the following model architectures:

ALBERT, BART, BERT, BigBird, BigBird-Pegasus, BioGpt, BLOOM, CamemBERT, CANINE, ConvBERT, CTRL, Data2VecText, DeBERTa, DeBERTa-v2, DistilBERT, ELECTRA, ERNIE, ErnieM, ESM, FlauBERT, FNet, Funnel Transformer, GPT-Sw3, OpenAI GPT-2, GPTBigCode, GPT Neo, GPT NeoX, GPT-J, I-BERT, LayoutLM, LayoutLMv2, LayoutLMv3, LED, LiLT, LLaMA, Longformer, LUKE, MarkupLM, mBART, MEGA, Megatron-BERT, MobileBERT, MPNet, MVP, Nezha, Nyströmformer, OpenLlama, OpenAI GPT, OPT, Perceiver, PLBart, QDQBert, Reformer, RemBERT, RoBERTa, RoBERTa-PreLayerNorm, RoCBert, RoFormer, SqueezeBERT, TAPAS, Transformer-XL, XLM, XLM-RoBERTa, XLM-RoBERTa-XL, XLNet, X-MOD, YOSO

Before you begin, make sure you have all the necessary libraries installed:

```
pip install transformers datasets evaluate
```

We encourage you to login to your Hugging Face account so you can upload and share your model with the community. When prompted, enter your token to login:

```
!pip install transformers datasets evaluate
```

```
Requirement already satisfied: transformers in /usr/local/lib/python3.10/dist-packages (4.40.1)
Requirement already satisfied: datasets in /usr/local/lib/python3.10/dist-packages (2.19.0)
Requirement already satisfied: evaluate in /usr/local/lib/python3.10/dist-packages (0.4.1)
Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-packages (from transformers) (3.13.4)
Requirement already satisfied: huggingface-hub<1.0,>=0.19.3 in /usr/local/lib/python3.10/dist-packages (from transformers) (
Requirement already satisfied: numpy>=1.17 in /usr/local/lib/python3.10/dist-packages (from transformers) (1.25.2)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.10/dist-packages (from transformers) (24.0)
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.10/dist-packages (from transformers) (6.0.1)
Requirement already satisfied: regex!=2019.12.17 in /usr/local/lib/python3.10/dist-packages (from transformers) (2023.12.25)
Requirement already satisfied: requests in /usr/local/lib/python3.10/dist-packages (from transformers) (2.31.0)
Requirement already satisfied: tokenizers<0.20,>=0.19 in /usr/local/lib/python3.10/dist-packages (from transformers) (0.19.1
Requirement already satisfied: safetensors>=0.4.1 in /usr/local/lib/python3.10/dist-packages (from transformers) (0.4.3)
Requirement already satisfied: tqdm>=4.27 in /usr/local/lib/python3.10/dist-packages (from transformers) (4.66.2)
Requirement already satisfied: pyarrow>=12.0.0 in /usr/local/lib/python3.10/dist-packages (from datasets) (14.0.2)
Requirement already satisfied: pyarrow-hotfix in /usr/local/lib/python3.10/dist-packages (from datasets) (0.6)
Requirement already satisfied: dill<0.3.9,>=0.3.0 in /usr/local/lib/python3.10/dist-packages (from datasets) (0.3.8)
Requirement already satisfied: pandas in /usr/local/lib/python3.10/dist-packages (from datasets) (2.0.3)
Requirement already satisfied: xxhash in /usr/local/lib/python3.10/dist-packages (from datasets) (3.4.1)
Requirement already satisfied: multiprocess in /usr/local/lib/python3.10/dist-packages (from datasets) (0.70.16)
Requirement already satisfied: fsspec[http]<=2024.3.1,>=2023.1.0 in /usr/local/lib/python3.10/dist-packages (from datasets)
Requirement already satisfied: aiohttp in /usr/local/lib/python3.10/dist-packages (from datasets) (3.9.5)
Requirement already satisfied: responses<0.19 in /usr/local/lib/python3.10/dist-packages (from evaluate) (0.18.0)
Requirement already satisfied: aiosignal>=1.1.2 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (1.3.1)
Requirement already satisfied: attrs>=17.3.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (23.2.0)
Requirement already satisfied: frozenlist>=1.1.1 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (1.4.1)
Requirement already satisfied: multidict<7.0,>=4.5 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (6.0.
Requirement already satisfied: yarl<2.0,>=1.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (1.9.4)
Requirement already satisfied: async-timeout<5.0,>=4.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (
Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/lib/python3.10/dist-packages (from huggingface-hub<1
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests->transform
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests->transformers) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests->transformers) (
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests->transformers) (
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.10/dist-packages (from pandas->datasets) (2.
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-packages (from pandas->datasets) (2023.4)
Requirement already satisfied: tzdata>=2022.1 in /usr/local/lib/python3.10/dist-packages (from pandas->datasets) (2024.1)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.8.2->pandas->dat
```

```
from huggingface_hub import notebook_login

notebook_login()
```

```
Token is valid (permission: write).
en has been saved in your configured git credential helpers
our token has been saved to /root/.cache/huggingface/toke
Login successful
```

Load IMDb dataset

Start by loading the IMDb dataset from the 🤗 Datasets library:

```
from datasets import load_dataset

imdb = load_dataset("imdb")
```

```
/usr/local/lib/python3.10/dist-packages/huggingface_hub/utils/_token.py:89: 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), set
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 models or datasets.
  warnings.warn(
```

Then take a look at an example:

```
imdb["test"][0]
```

```
{'text': 'I love sci-fi and am willing to put up with a lot. Sci-fi movies/TV are usually underfunded, under-appreciated
and misunderstood. I tried to like this, I really did, but it is to good TV sci-fi as Babylon 5 is to Star Trek (the
original). Silly prosthetics, cheap cardboard sets, stilted dialogues, CG that doesn\'t match the background, and painfully
one-dimensional characters cannot be overcome with a \'sci-fi\' setting. (I\'m sure there are those of you out there who
think Babylon 5 is good sci-fi TV. It\'s not. It\'s clichéd and uninspiring.) While US viewers might like emotion and
character development, sci-fi is a genre that does not take itself seriously (cf. Star Trek). It may treat important
issues, yet not as a serious philosophy. It\'s really difficult to care about the characters here as they are not simply
foolish, just missing a spark of life. Their actions and reactions are wooden and predictable, often painful to watch. The
makers of Earth KNOW it\'s rubbish as they have to always say "Gene Roddenberry\'s Earth..." otherwise people would not
continue watching. Roddenberry\'s ashes must be turning in their orbit as this dull, cheap, poorly edited (watching it
without advert breaks really brings this home) trudging Trabant of a show lumbers into space. Spoiler. So, kill off a main
character. And then bring him back as another actor. Jeeez! Dallas all over again.',
 'label': 0}
```

There are two fields in this dataset:

- `text`: the movie review text.
- `label`: a value that is either `0` for a negative review or `1` for a positive review.

Preprocess

The next step is to load a DistilBERT tokenizer to preprocess the `text` field:

```
from transformers import AutoTokenizer

tokenizer = AutoTokenizer.from_pretrained("distilbert-base-uncased")
```

Create a preprocessing function to tokenize `text` and truncate sequences to be no longer than DistilBERT's maximum input length:

```
def preprocess_function(examples):
    return tokenizer(examples["text"], truncation=True, max_length=512, padding="max_length")
```

To apply the preprocessing function over the entire dataset, use 🤗 Datasets map function. You can speed up `map` by setting `batched=True` to process multiple elements of the dataset at once:

> Add blockquote

```
tokenized_imdb = imdb.map(preprocess_function, batched=True)
```

```
Map: 100%                                25000/25000 [00:07<00:00, 3585.52 examples/s]
```

Now create a batch of examples using DataCollatorWithPadding. It's more efficient to *dynamically pad* the sentences to the longest length in a batch during collation, instead of padding the whole dataset to the maximum length.

```
from transformers import DataCollatorWithPadding

data_collator = DataCollatorWithPadding(tokenizer=tokenizer)
```

Evaluate

Including a metric during training is often helpful for evaluating your model's performance. You can quickly load a evaluation method with the 🤗 Evaluate library. For this task, load the accuracy metric (see the 🤗 Evaluate quick tour to learn more about how to load and compute a metric):

```
import evaluate

accuracy = evaluate.load("accuracy")
```

Then create a function that passes your predictions and labels to compute to calculate the accuracy:

```
import numpy as np


def compute_metrics(eval_pred):
    predictions, labels = eval_pred
    predictions = np.argmax(predictions, axis=1)
    return accuracy.compute(predictions=predictions, references=labels)
```

Your `compute_metrics` function is ready to go now, and you'll return to it when you setup your training.

Train

Before you start training your model, create a map of the expected ids to their labels with `id2label` and `label2id`:

```
id2label = {0: "NEGATIVE", 1: "POSITIVE"}
label2id = {"NEGATIVE": 0, "POSITIVE": 1}
```

If you aren't familiar with finetuning a model with the Trainer, take a look at the basic tutorial here!

You're ready to start training your model now! Load DistilBERT with AutoModelForSequenceClassification along with the number of expected labels, and the label mappings:

```
from transformers import AutoModelForSequenceClassification, TrainingArguments, Trainer

model = AutoModelForSequenceClassification.from_pretrained(
    "distilbert-base-uncased", num_labels=2, id2label=id2label, label2id=label2id
)
```

```
Some weights of DistilBertForSequenceClassification were not initialized from the model checkpoint at distilbert-base-uncase
You should probably TRAIN this model on a down-stream task to be able to use it for predictions and inference.
```

At this point, only three steps remain:

1. Define your training hyperparameters in TrainingArguments. The only required parameter is `output_dir` which specifies where to save your model. You'll push this model to the Hub by setting `push_to_hub=True` (you need to be signed in to Hugging Face to upload your model). At the end of each epoch, the Trainer will evaluate the accuracy and save the training checkpoint.
2. Pass the training arguments to Trainer along with the model, dataset, tokenizer, data collator, and `compute_metrics` function.
3. Call train() to finetune your model.

```
!pip install accelerate -U
!pip install transformers[torch] -U
import accelerate
import transformers
training_args = TrainingArguments(
    output_dir="my_awesome_model",
    learning_rate=2e-5,
    per_device_train_batch_size=16,
    per_device_eval_batch_size=16,
    num_train_epochs=2,
    weight_decay=0.01,
    evaluation_strategy="epoch",
    save_strategy="epoch",
    load_best_model_at_end=True,
    push_to_hub=True,
)

trainer = Trainer(
    model=model,
    args=training_args,
    train_dataset=tokenized_imdb["train"],
    eval_dataset=tokenized_imdb["test"],
    tokenizer=tokenizer,
    data_collator=data_collator,
    compute_metrics=compute_metrics,
)

trainer.train()
```

```
Requirement already satisfied: accelerate in /usr/local/lib/python3.10/dist-pack
Requirement already satisfied: numpy>=1.17 in /usr/local/lib/python3.10/dist-pac
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.10/dist
Requirement already satisfied: psutil in /usr/local/lib/python3.10/dist-packages
Requirement already satisfied: pyyaml in /usr/local/lib/python3.10/dist-packages
Requirement already satisfied: torch>=1.10.0 in /usr/local/lib/python3.10/dist-p
Requirement already satisfied: huggingface-hub in /usr/local/lib/python3.10/dist
Requirement already satisfied: safetensors>=0.3.1 in /usr/local/lib/python3.10/d
Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-packag
Requirement already satisfied: typing-extensions>=4.8.0 in /usr/local/lib/python
Requirement already satisfied: sympy in /usr/local/lib/python3.10/dist-packages
Requirement already satisfied: networkx in /usr/local/lib/python3.10/dist-packag
Requirement already satisfied: jinja2 in /usr/local/lib/python3.10/dist-packages
Requirement already satisfied: fsspec in /usr/local/lib/python3.10/dist-packages
Requirement already satisfied: nvidia-cuda-nvrtc-cu12==12.1.105 in /usr/local/li
Requirement already satisfied: nvidia-cuda-runtime-cu12==12.1.105 in /usr/local/
Requirement already satisfied: nvidia-cuda-cupti-cu12==12.1.105 in /usr/local/li
Requirement already satisfied: nvidia-cudnn-cu12==8.9.2.26 in /usr/local/lib/pyt
Requirement already satisfied: nvidia-cublas-cu12==12.1.3.1 in /usr/local/lib/py
Requirement already satisfied: nvidia-cufft-cu12==11.0.2.54 in /usr/local/lib/py
Requirement already satisfied: nvidia-curand-cu12==10.3.2.106 in /usr/local/lib/
Requirement already satisfied: nvidia-cusolver-cu12==11.4.5.107 in /usr/local/li
Requirement already satisfied: nvidia-cusparse-cu12==12.1.0.106 in /usr/local/li
Requirement already satisfied: nvidia-nccl-cu12==2.19.3 in /usr/local/lib/python
Requirement already satisfied: nvidia-nvtx-cu12==12.1.105 in /usr/local/lib/pyth
Requirement already satisfied: triton==2.2.0 in /usr/local/lib/python3.10/dist-p
Requirement already satisfied: nvidia-nvjitlink-cu12 in /usr/local/lib/python3.1
Requirement already satisfied: requests in /usr/local/lib/python3.10/dist-packag
Requirement already satisfied: tqdm>=4.42.1 in /usr/local/lib/python3.10/dist-pa
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.10/dist
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-pa
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/d
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/d
Requirement already satisfied: mpmath>=0.19 in /usr/local/lib/python3.10/dist-pa
Requirement already satisfied: transformers[torch] in /usr/local/lib/python3.10/
Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-packag
Requirement already satisfied: huggingface-hub<1.0,>=0.19.3 in /usr/local/lib/py
Requirement already satisfied: numpy>=1.17 in /usr/local/lib/python3.10/dist-pac
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.10/dist
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.10/dist-pac
Requirement already satisfied: regex!=2019.12.17 in /usr/local/lib/python3.10/di
Requirement already satisfied: requests in /usr/local/lib/python3.10/dist-packag
Requirement already satisfied: tokenizers<0.20,>=0.19 in /usr/local/lib/python3.
Requirement already satisfied: safetensors>=0.4.1 in /usr/local/lib/python3.10/d
Requirement already satisfied: tqdm>=4.27 in /usr/local/lib/python3.10/dist-pack
Requirement already satisfied: torch in /usr/local/lib/python3.10/dist-packages
Requirement already satisfied: accelerate>=0.21.0 in /usr/local/lib/python3.10/d
Requirement already satisfied: psutil in /usr/local/lib/python3.10/dist-packages
Requirement already satisfied: fsspec>=2023.5.0 in /usr/local/lib/python3.10/dis
Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/lib/pyth
Requirement already satisfied: sympy in /usr/local/lib/python3.10/dist-packages
Requirement already satisfied: networkx in /usr/local/lib/python3.10/dist-packag
Requirement already satisfied: jinja2 in /usr/local/lib/python3.10/dist-packages
Requirement already satisfied: nvidia-cuda-nvrtc-cu12==12.1.105 in /usr/local/li
Requirement already satisfied: nvidia-cuda-runtime-cu12==12.1.105 in /usr/local/
Requirement already satisfied: nvidia-cuda-cupti-cu12==12.1.105 in /usr/local/li
Requirement already satisfied: nvidia-cudnn-cu12==8.9.2.26 in /usr/local/lib/pyt
Requirement already satisfied: nvidia-cublas-cu12==12.1.3.1 in /usr/local/lib/py
Requirement already satisfied: nvidia-cufft-cu12==11.0.2.54 in /usr/local/lib/py
Requirement already satisfied: nvidia-curand-cu12==10.3.2.106 in /usr/local/lib/
Requirement already satisfied: nvidia-cusolver-cu12==11.4.5.107 in /usr/local/li
Requirement already satisfied: nvidia-cusparse-cu12==12.1.0.106 in /usr/local/li
Requirement already satisfied: nvidia-nccl-cu12==2.19.3 in /usr/local/lib/python
Requirement already satisfied: nvidia-nvtx-cu12==12.1.105 in /usr/local/lib/pyth
Requirement already satisfied: triton==2.2.0 in /usr/local/lib/python3.10/dist-p
Requirement already satisfied: nvidia-nvjitlink-cu12 in /usr/local/lib/python3.1
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-pa
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/d
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/d
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.10/dist
Requirement already satisfied: mpmath>=0.19 in /usr/local/lib/python3.10/dist-pa
```

[3126/3126 29:43, Epoch 2/2]

Epoch	Training Loss	Validation Loss	Accuracy
1	0.224400	0.206425	0.920680
2	0.138800	0.232338	0.931840

```
TrainOutput(global_step=3126, training_loss=0.20436020654054765, metrics=
{'train_runtime': 1785.6368, 'train_samples_per_second': 28.001,
```

Trainer applies dynamic padding by default when you pass `tokenizer` to it. In this case, you don't need to specify a data collator explicitly.

Once training is completed, share your model to the Hub with the push_to_hub() method so everyone can use your model:

```
trainer.push_to_hub()
```

```
    CommitInfo(commit_url='https://huggingface.co/GauravR12060102/my_awesome_model/c
    commit_message='End of training', commit_description='',
```

For a more in-depth example of how to finetune a model for text classification, take a look at the corresponding PyTorch notebook or TensorFlow notebook.

Inference

Great, now that you've finetuned a model, you can use it for inference!

Grab some text you'd like to run inference on:

```
text = "This was a masterpiece. Not completely faithful to the books, but enthralling from beginning to end. Might be my favorit
```