Hugging Face Transformers: A Step-by-Step Guide

DECODING DATA SCIENCE .com
Community
Consulting
Academy
Events

Objective:

To provide a structured, beginner-friendly, and code-rich walkthrough of Hugging Face's Transformers library covering:

- Installation
- Using pipelines
- Loading models/tokenizers
- Fine-tuning
- Saving/loading models
- Training with Trainer and TensorFlow

Installation & Setup

Choose Framework

```
pip install transformers datasets evaluate
accelerate

#PyTorch:
pip install torch

#TensorFlow:
pip install tensorflow
```

Use Pretrained Models with pipeline()

💡 **Sentiment Analysis Example:**

```
from transformers import pipeline

classifier = pipeline("sentiment-analysis")
print(classifier("We are very happy to show you the 🤗 Transformers library."))
```

Multiple Inputs:

```
texts = ["Love this!", "Hate that."]
results = classifier(texts)
for result in results:
    print(f"{result['label']}
({round(result['score'], 2)})")
```

✅ 3. Speech Recognition Example

```
from transformers import pipeline
from datasets import load_dataset, Audio

asr = pipeline("automatic-speech-recognition",
model="facebook/wav2vec2-base-960h")
dataset = load_dataset("PolyAI/minds14", name="en-US",
split="train")
dataset = dataset.cast_column("audio",
Audio(sampling_rate=asr.feature_extractor.sampling_rat
e))

audio_inputs = dataset[:4]["audio"]
texts = asr(audio_inputs)
print([t["text"] for t in texts])
```

✅ 4. Using Custom Model and Tokenizer

```
from transformers import pipeline,
AutoModelForSequenceClassification,
AutoTokenizer

model_name = "nlptown/bert-base-multilingual-
uncased-sentiment"
model =
AutoModelForSequenceClassification.from_pretrai
ned(model_name)
tokenizer =
AutoTokenizer.from_pretrained(model_name)
```

✅ 4. Using Custom Model and Tokenizer

```
from transformers import pipeline,
AutoModelForSequenceClassification,
AutoTokenizer

model_name = "nlptown/bert-base-multilingual-
uncased-sentiment"
model =
AutoModelForSequenceClassification.from_pretrai
ned(model_name)
tokenizer =
AutoTokenizer.from_pretrained(model_name)
```

5. AutoTokenizer and AutoModel

```
from transformers import AutoTokenizer,
AutoModelForSequenceClassification
import torch.nn.functional as F

tokenizer =
AutoTokenizer.from_pretrained(model_name)
model =
AutoModelForSequenceClassification.from_pretrained(m
odel_name)

inputs = tokenizer(["Amazing!", "Terrible."],
return_tensors="pt", padding=True, truncation=True)
outputs = model(**inputs)
probs = F.softmax(outputs.logits, dim=-1)
print(probs)
```

6. Saving and Loading Models

```
# Save
model.save_pretrained("./my_model")
tokenizer.save_pretrained("./my_model")

# Load
from transformers import
AutoModelForSequenceClassification, AutoTokenizer
model =
AutoModelForSequenceClassification.from_pretrained("
./my_model")
tokenizer =
AutoTokenizer.from_pretrained("./my_model")
```

What Next? Join the Free AI Community

Artificial Intelligence

2,810 members

SCAN ME

Free

- Three weekly events
- Live workshops
- Knowledge Shorts 50+ Videos
- Basic AI & DS courses
- DS & AI materials
- Webinar recording
- Guidance from experts
- 24 by 7 Whatsapp & Discord
- Latest ai Discussion & More...

nas.io/artificialintelligence

DECODING DATA SCIENCE

Community Profile

SCAN ME

What Does The Community Provide?

Generative AI

- ✅ Generative AI (chatGPT) for Business
- ✅ Prompt Engineering for Developers
- ✅ Langchain for AI App Development
- ✅ What are Large Language Models
- ✅ Gemini MultiModal LLM

Recordings

- ✅ 50+ Outcome-based Workshops
- ✅ 45+ AI Community Meetup Recordings
- ✅ 10+ Python Projects Videos
- ✅ AI & DS Career & Learning Webinar Series

Data Science Courses

- ✅ Basic Excel For Data Science
- ✅ Basic SQL For AI/Data Science
- ✅ Basic Python for AI/Data Jobs
- ✅ Advanced Python for AI/DS Jobs
- ✅ Basic PowerBI for AI/Data Science
- ✅ Machine Learning
- ✅ Knowledge Shorts
- ✅ Linkedin Growth Mastery

Resources

- ✅ Generative AI Resources
- ✅ Sample Datasets & Projects
- ✅ Sample Reviewed Resume
- ✅ Ready to use Resume Template
- ✅ Linkedin Profile Optimization
- ✅ Essential SQL Documents
- ✅ Essential Python Documents
- ✅ Machine Learning Documents

Every week we have live Zoom calls, Physical Meetups and LinkedIn Audio events and WhatsApp discussions. All calls are recorded and archived.

nas.io/aiguild