Intro Hugging Face and Python

Dirk Wulff & Zak Hussain

MAX PLANCK INSTITUTE
FOR HUMAN DEVELOPMENT

UNI
BASEL

OUR SOFTWARE STACK

+

+

OUR SOFTWARE STACK

+

+

Google
colab

WHY IS HUGGING FACE? 🤗

WHY IS HUGGING FACE? 🤗

Traditional language modelling pipeline:

WHY IS HUGGING FACE? 🤗

Traditional language modelling pipeline:

1. Find out the model architecture

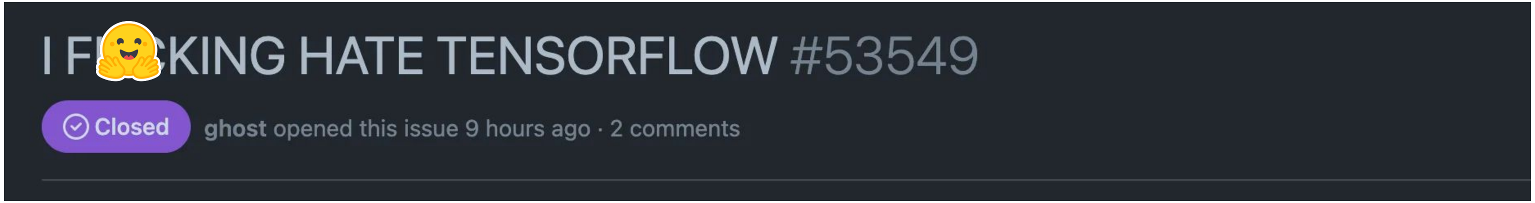
WHY IS HUGGING FACE? 🤗

Traditional language modelling pipeline:

1. Find out the model architecture
2. Implement the model architecture in code with deep learning frameworks (e.g PyTorch/Tensorflow).

1. DEEP LEARNING LIBRARIES CAN BE DIFFICULT

1. DEEP LEARNING LIBRARIES CAN BE DIFFICULT



I F🤗CKING HATE TENSORFLOW #53549

✓ Closed ghost opened this issue 9 hours ago · 2 comments

WHY HUGGING FACE?

Traditional language modelling pipeline:

1. Find out the model architecture
2. Implement the model architecture in code with deep learning libraries (e.g PyTorch/Tensorflow).

WHY HUGGING FACE?

Traditional language modelling pipeline:

1. Find out the model architecture
2. Implement the model architecture in code with deep learning libraries (e.g PyTorch/Tensorflow).
3. Load the pretrained weights (if available) from a server.
4. Process the inputs (using the correct tokenizer for the model)
5. Implement data loaders
6. Define a loss function
7. Stick a task-specific “head” on the model

HUGGING FACE PIPELINES

Import pipeline

```
from transformers import pipeline
```

Initialise pipeline

```
pipe = pipeline('text-generation', model='gpt2')
```

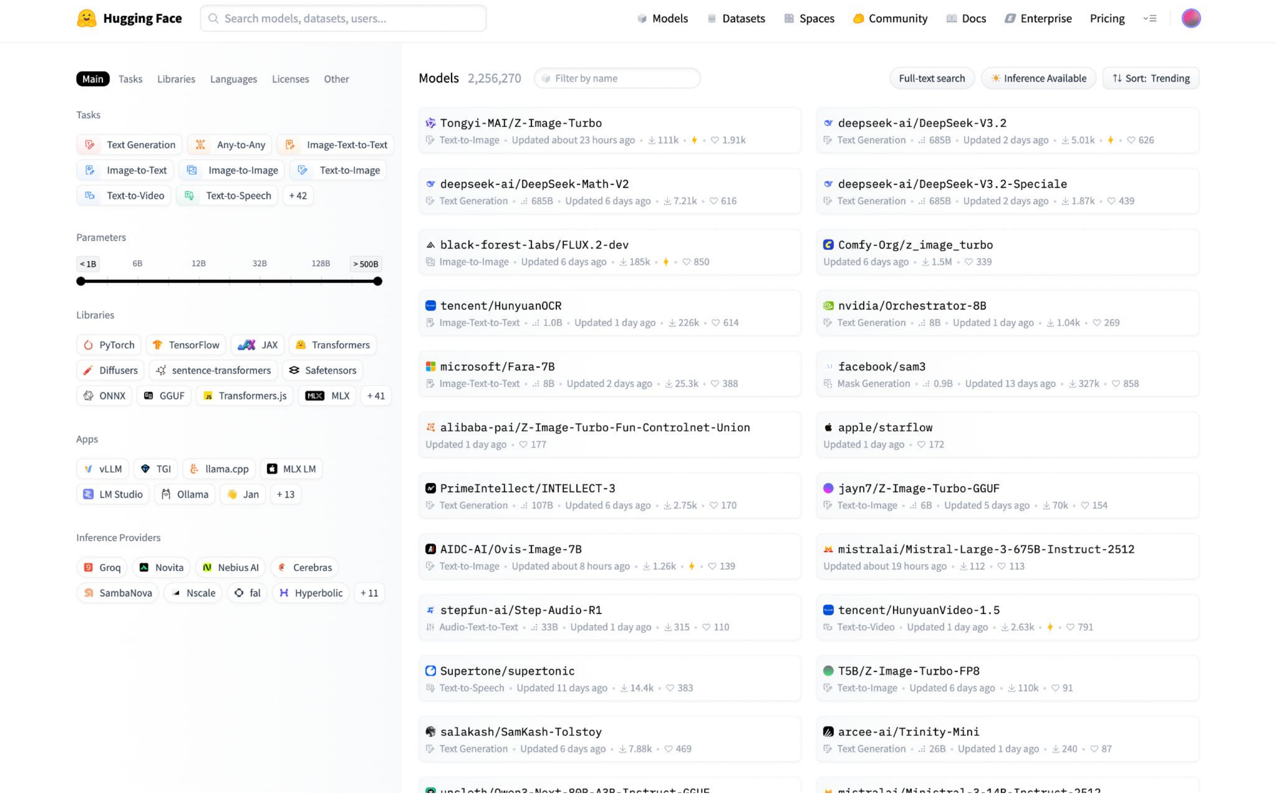
Load model input

```
prompt = """
Once upon a time in a land far far away, there was a young prince named
John. He was known for his bravery and courage. One day, he decided to go on
an adventure to explore the unknown lands.
"""`
```

Feed input the model

```
output = pipe(prompt, max_length=100)
print(output)
```

HUGGING FACE



Hugging Face · Search models, datasets, users...

Models · Datasets · Spaces · Community · Docs · Enterprise · Pricing

Main · Tasks · Libraries · Languages · Licenses · Other

Tasks

Text Generation · Any-to-Any · Image-Text-to-Text · Image-to-Text · Image-to-Image · Text-to-Image · Text-to-Video · Text-to-Speech · + 42

Parameters

< 1B · 6B · 12B · 32B · 128B · > 500B

Libraries

PyTorch · TensorFlow · JAX · Transformers · Diffusers · sentence-transformers · Safetensors · ONNX · GGUF · Transformers.js · MLX · + 41

Apps

vLLM · TGI · llama.cpp · MLX LM · LM Studio · Ollama · Jan · + 13

Inference Providers

Groq · Novita · Nebius AI · Cerebras · SambaNova · Nscale · fal · Hyperbolic · + 11

Models 2,256,270 · Filter by name · Full-text search · Inference Available · Sort: Trending

Tongyi-MAI/Z-Image-Turbo
Text-to-Image · Updated about 23 hours ago · 111k · 1.91k

deepseek-ai/DeepSeek-V3.2
Text Generation · 685B · Updated 2 days ago · 5.01k · 626

deepseek-ai/DeepSeek-Math-V2
Text Generation · 685B · Updated 6 days ago · 7.21k · 616

deepseek-ai/DeepSeek-V3.2-Speciale
Text Generation · 685B · Updated 2 days ago · 1.87k · 439

black-forest-labs/FLUX.2-dev
Image-to-Image · Updated 6 days ago · 185k · 850

Comfy-Org/z_image_turbo
Updated 6 days ago · 1.5M · 339

tencent/HunyuanOCR
Image-Text-to-Text · 1.0B · Updated 1 day ago · 226k · 614

nvidia/Orchestrator-8B
Text Generation · 8B · Updated 1 day ago · 1.04k · 269

microsoft/Fara-7B
Image-Text-to-Text · 8B · Updated 2 days ago · 25.3k · 388

facebook/sam3
Mask Generation · 0.9B · Updated 13 days ago · 327k · 858

alibaba-pai/Z-Image-Turbo-Fun-Controlnet-Union
Updated 1 day ago · 177

apple/starflow
Updated 1 day ago · 172

PrimeIntellect/INTELLECT-3
Text Generation · 107B · Updated 6 days ago · 2.75k · 170

jayn7/Z-Image-Turbo-GGUF
Text-to-Image · 6B · Updated 5 days ago · 70k · 154

AIDC-AI/Ovis-Image-7B
Text-to-Image · Updated about 8 hours ago · 1.26k · 139

mistralai/Mistral-Large-3-675B-Instruct-2512
Updated about 19 hours ago · 112 · 113

stepfun-ai/Step-Audio-R1
Audio-Text-to-Text · 33B · Updated 1 day ago · 315 · 110

tencent/HunyuanVideo-1.5
Text-to-Video · Updated 1 day ago · 2.63k · 791

Supertone/supertonic
Text-to-Speech · Updated 11 days ago · 14.4k · 383

T5B/Z-Image-Turbo-FP8
Text-to-Image · Updated 6 days ago · 110k · 91

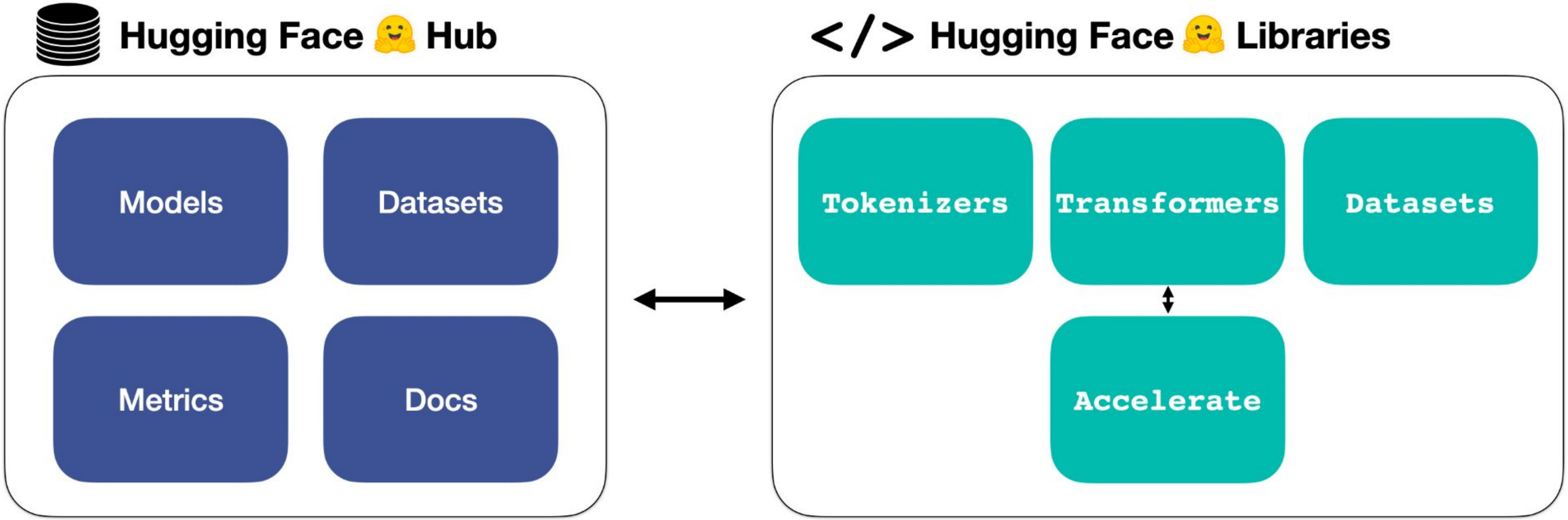
salakash/SamKash-Tolstoy
Text Generation · Updated 6 days ago · 7.88k · 469

arcee-ai/Trinity-Mini
Text Generation · 26B · Updated 1 day ago · 240 · 87

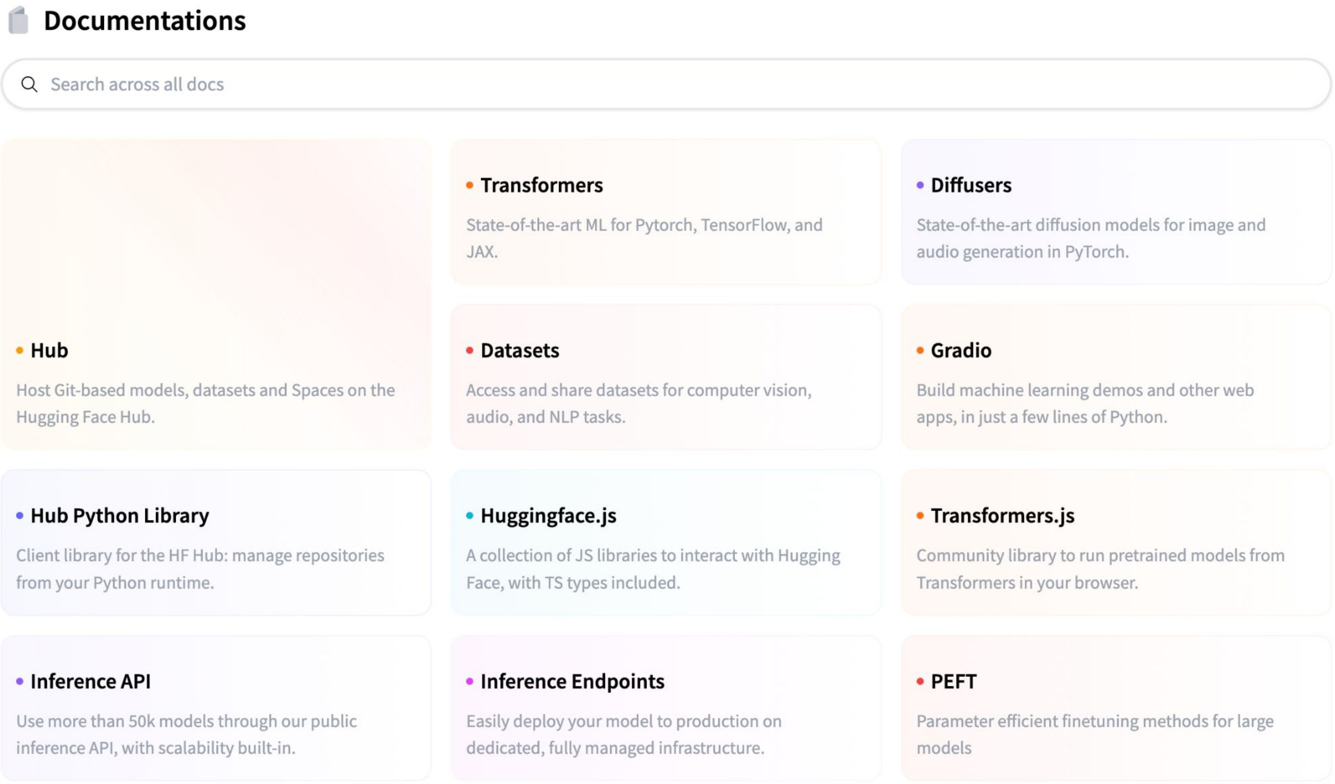
unsloth/Qwen3-Next-80B-A3B-Instruct-GGUF

mistralai/Ministral-3-14B-Instruct-2512

THE HUGGING FACE ECOSYSTEM



HUGGING FACE DOCUMENTATION



Documentations

Search across all docs

- **Hub**
  Host Git-based models, datasets and Spaces on the Hugging Face Hub.
- **Transformers**
  State-of-the-art ML for Pytorch, TensorFlow, and JAX.
- **Diffusers**
  State-of-the-art diffusion models for image and audio generation in PyTorch.
- **Datasets**
  Access and share datasets for computer vision, audio, and NLP tasks.
- **Gradio**
  Build machine learning demos and other web apps, in just a few lines of Python.
- **Hub Python Library**
  Client library for the HF Hub: manage repositories from your Python runtime.
- **Huggingface.js**
  A collection of JS libraries to interact with Hugging Face, with TS types included.
- **Transformers.js**
  Community library to run pretrained models from Transformers in your browser.
- **Inference API**
  Use more than 50k models through our public inference API, with scalability built-in.
- **Inference Endpoints**
  Easily deploy your model to production on dedicated, fully managed infrastructure.
- **PEFT**
  Parameter efficient finetuning methods for large models

OUR SOFTWARE STACK

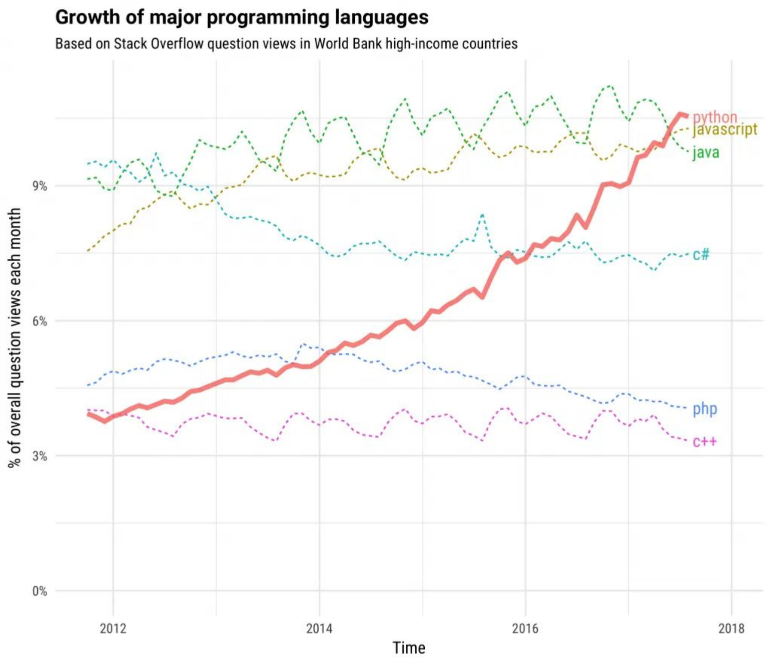
+

+

PYTHON

PYTHON

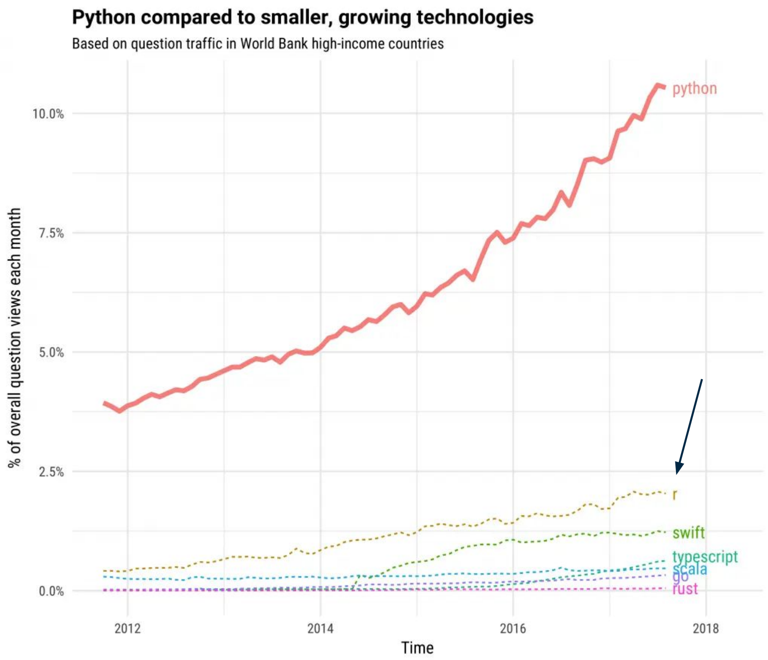
- A **high-level** programming language known for its simplicity and readability.
- Used in various domains such as web development, data analysis, **artificial intelligence**, and scientific computing.


Growth of major programming languages

Based on Stack Overflow question views in World Bank high-income countries

PYTHON

- A **high-level** programming language known for its simplicity and readability.
- Used in various domains such as web development, data analysis, **artificial intelligence**, and scientific computing.



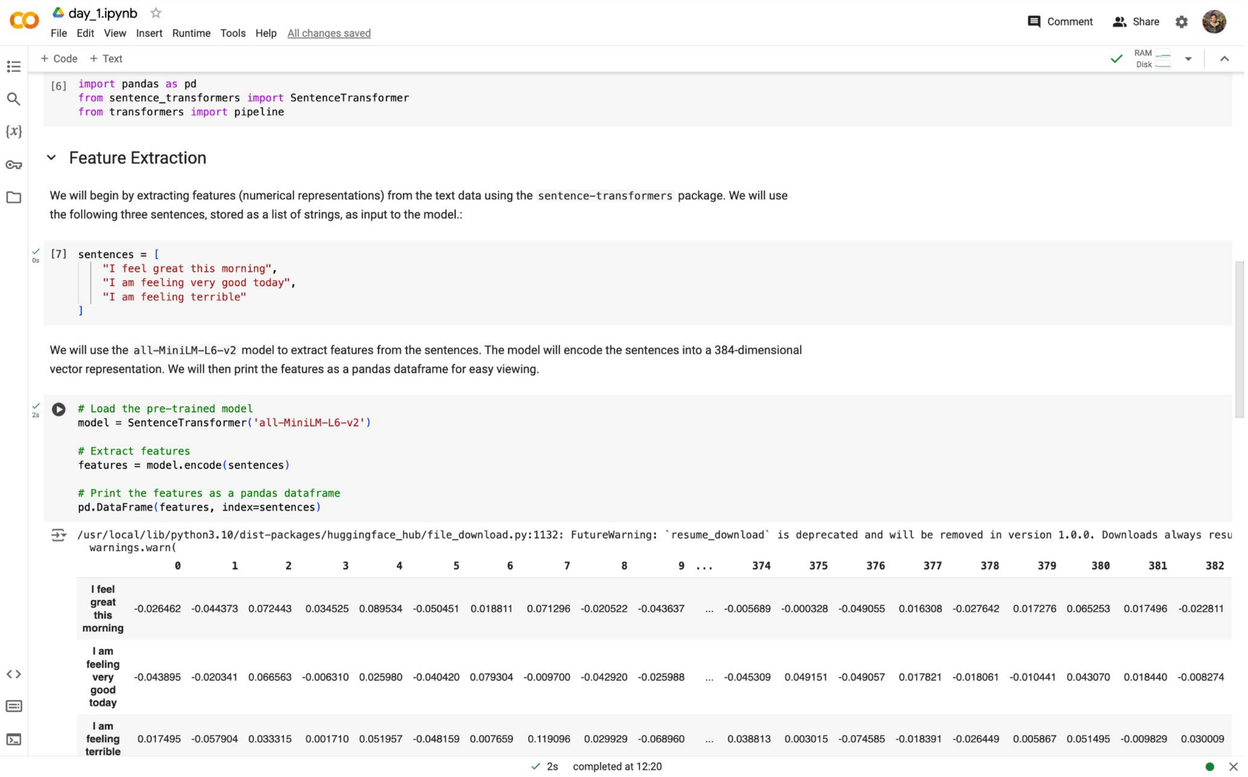
OUR SOFTWARE STACK

+

+

PYTHON + Google Colab

PYTHON + Google Colab



```
[6] import pandas as pd
    from sentence_transformers import SentenceTransformer
    from transformers import pipeline
```

Feature Extraction

We will begin by extracting features (numerical representations) from the text data using the `sentence-transformers` package. We will use the following three sentences, stored as a list of strings, as input to the model.:

```
[7] sentences = [
        "I feel great this morning",
        "I am feeling very good today",
        "I am feeling terrible"
    ]
```

We will use the `all-MiniLM-L6-v2` model to extract features from the sentences. The model will encode the sentences into a 384-dimensional vector representation. We will then print the features as a pandas dataframe for easy viewing.

```
# Load the pre-trained model
model = SentenceTransformer('all-MiniLM-L6-v2')

# Extract features
features = model.encode(sentences)

# Print the features as a pandas dataframe
pd.DataFrame(features, index=sentences)
```

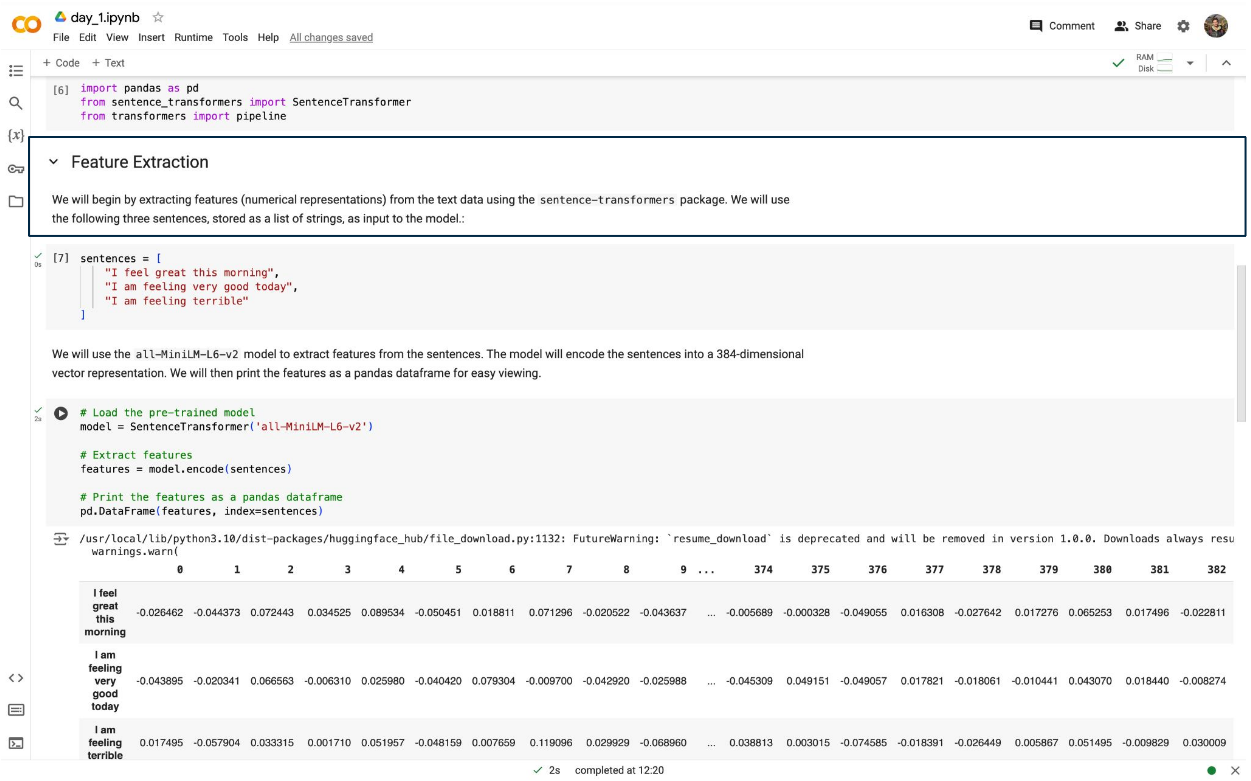
```
/usr/local/lib/python3.10/dist-packages/huggingface_hub/file_download.py:1132: FutureWarning: `resume_download` is deprecated and will be removed in version 1.0.0. Downloads always resu
  warnings.warn(
```

	0	1	2	3	4	5	6	7	8	9	...	374	375	376	377	378	379	380	381	382
I feel great this morning	-0.026462	-0.044373	0.072443	0.034525	0.089534	-0.050451	0.018811	0.071296	-0.020522	-0.043637	...	-0.005689	-0.000328	-0.049055	0.016308	-0.027642	0.017276	0.065253	0.017496	-0.022811
I am feeling very good today	-0.043895	-0.020341	0.066563	-0.006310	0.025980	-0.040420	0.079304	-0.009700	-0.042920	-0.025988	...	-0.045309	0.049151	-0.049057	0.017821	-0.018061	-0.010441	0.043070	0.018440	-0.008274
I am feeling terrible	0.017495	-0.057904	0.033315	0.001710	0.051957	-0.048159	0.007659	0.119096	0.029929	-0.068960	...	0.038813	0.003015	-0.074585	-0.018391	-0.026449	0.005867	0.051495	-0.009829	0.030009

✓ 2s completed at 12:20

PYTHON + Google Colab

1. Markdown



```
[6] import pandas as pd
    from sentence_transformers import SentenceTransformer
    from transformers import pipeline
```

Feature Extraction

We will begin by extracting features (numerical representations) from the text data using the `sentence-transformers` package. We will use the following three sentences, stored as a list of strings, as input to the model.:

```
[7] sentences = [
        "I feel great this morning",
        "I am feeling very good today",
        "I am feeling terrible"
    ]
```

We will use the `all-MiniLM-L6-v2` model to extract features from the sentences. The model will encode the sentences into a 384-dimensional vector representation. We will then print the features as a pandas dataframe for easy viewing.

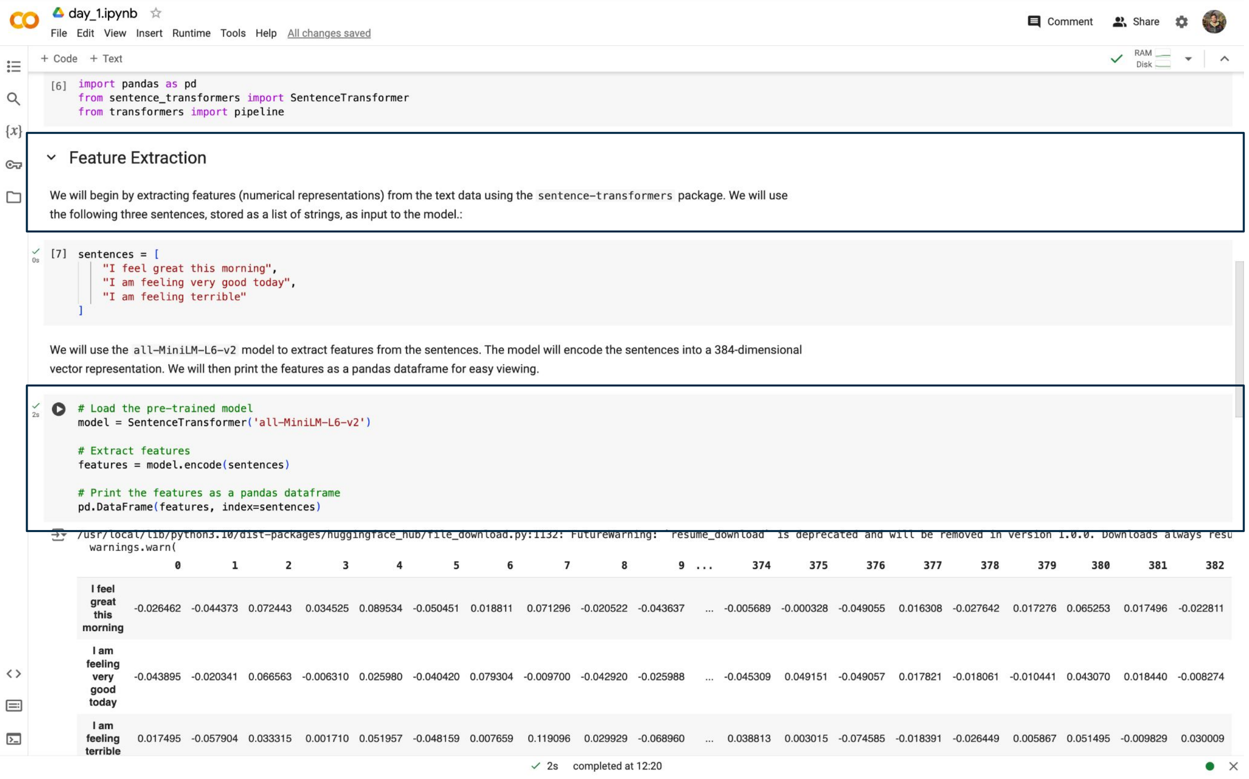
```
# Load the pre-trained model
model = SentenceTransformer('all-MiniLM-L6-v2')

# Extract features
features = model.encode(sentences)

# Print the features as a pandas dataframe
pd.DataFrame(features, index=sentences)
```

```
/usr/local/lib/python3.10/dist-packages/huggingface_hub/file_download.py:1132: FutureWarning: `resume_download` is deprecated and will be removed in version 1.0.0. Downloads always resu
  warnings.warn(
```

	0	1	2	3	4	5	6	7	8	9	...	374	375	376	377	378	379	380	381	382
I feel great this morning	-0.026462	-0.044373	0.072443	0.034525	0.089534	-0.050451	0.018811	0.071296	-0.020522	-0.043637	...	-0.005689	-0.000328	-0.049055	0.016308	-0.027642	0.017276	0.065253	0.017496	-0.022811
I am feeling very good today	-0.043895	-0.020341	0.066563	-0.006310	0.025980	-0.040420	0.079304	-0.009700	-0.042920	-0.025988	...	-0.045309	0.049151	-0.049057	0.017821	-0.018061	-0.010441	0.043070	0.018440	-0.008274
I am feeling terrible	0.017495	-0.057904	0.033315	0.001710	0.051957	-0.048159	0.007659	0.119096	0.029929	-0.068960	...	0.038813	0.003015	-0.074585	-0.018391	-0.026449	0.005867	0.051495	-0.009829	0.030009

✓ 2s completed at 12:20

PYTHON + Google Colab



```
[6] import pandas as pd
    from sentence_transformers import SentenceTransformer
    from transformers import pipeline
```

1. Markdown

Feature Extraction

We will begin by extracting features (numerical representations) from the text data using the `sentence-transformers` package. We will use the following three sentences, stored as a list of strings, as input to the model.:

```
[7] sentences = [
        "I feel great this morning",
        "I am feeling very good today",
        "I am feeling terrible"
    ]
```

We will use the `all-MiniLM-L6-v2` model to extract features from the sentences. The model will encode the sentences into a 384-dimensional vector representation. We will then print the features as a pandas dataframe for easy viewing.

2. Code

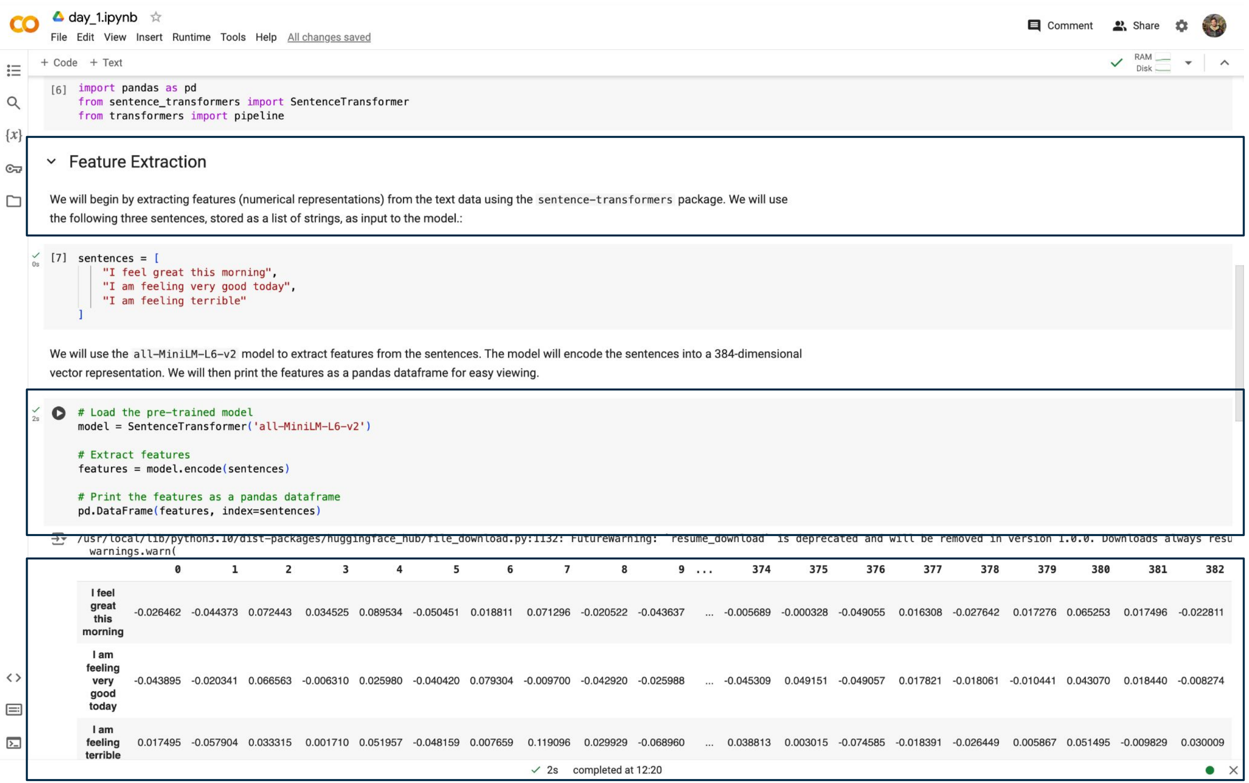
```
# Load the pre-trained model
model = SentenceTransformer('all-MiniLM-L6-v2')

# Extract features
features = model.encode(sentences)

# Print the features as a pandas dataframe
pd.DataFrame(features, index=sentences)
```

```
/usr/local/lib/python3.10/dist-packages/huggingface_hub/file_download.py:1132: FutureWarning: `resume_download` is deprecated and will be removed in version 1.0.0. Downloads always resu
  warnings.warn(
```

	0	1	2	3	4	5	6	7	8	9	...	374	375	376	377	378	379	380	381	382
I feel great this morning	-0.026462	-0.044373	0.072443	0.034525	0.089534	-0.050451	0.018811	0.071296	-0.020522	-0.043637	...	-0.005689	-0.000328	-0.049055	0.016308	-0.027642	0.017276	0.065253	0.017496	-0.022811
I am feeling very good today	-0.043895	-0.020341	0.066563	-0.006310	0.025980	-0.040420	0.079304	-0.009700	-0.042920	-0.025988	...	-0.045309	0.049151	-0.049057	0.017821	-0.018061	-0.010441	0.043070	0.018440	-0.008274
I am feeling terrible	0.017495	-0.057904	0.033315	0.001710	0.051957	-0.048159	0.007659	0.119096	0.029929	-0.068960	...	0.038813	0.003015	-0.074585	-0.018391	-0.026449	0.005867	0.051495	-0.009829	0.030009

✓ 2s completed at 12:20

PYTHON + Google Colab



```
[6] import pandas as pd
    from sentence_transformers import SentenceTransformer
    from transformers import pipeline
```

1. Markdown

Feature Extraction

We will begin by extracting features (numerical representations) from the text data using the `sentence-transformers` package. We will use the following three sentences, stored as a list of strings, as input to the model.:

```
[7] sentences = [
        "I feel great this morning",
        "I am feeling very good today",
        "I am feeling terrible"
    ]
```

We will use the `all-MiniLM-L6-v2` model to extract features from the sentences. The model will encode the sentences into a 384-dimensional vector representation. We will then print the features as a pandas dataframe for easy viewing.

2. Code

```
# Load the pre-trained model
model = SentenceTransformer('all-MiniLM-L6-v2')

# Extract features
features = model.encode(sentences)

# Print the features as a pandas dataframe
pd.DataFrame(features, index=sentences)
```

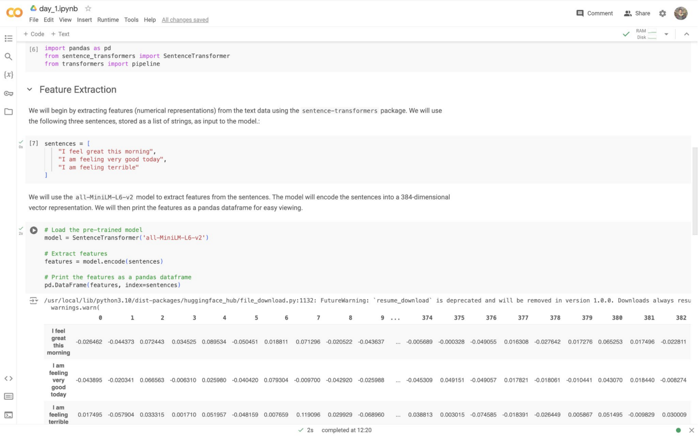
```
/usr/local/lib/python3.10/dist-packages/huggingface_hub/file_download.py:1132: FutureWarning: `resume_download` is deprecated and will be removed in version 1.0.0. Downloads always resu
  warnings.warn(
```

3. Printouts

	0	1	2	3	4	5	6	7	8	9	...	374	375	376	377	378	379	380	381	382
I feel great this morning	-0.026462	-0.044373	0.072443	0.034525	0.089534	-0.050451	0.018811	0.071296	-0.020522	-0.043637	...	-0.005689	-0.000328	-0.049055	0.016308	-0.027642	0.017276	0.065253	0.017496	-0.022811
I am feeling very good today	-0.043895	-0.020341	0.066563	-0.006310	0.025980	-0.040420	0.079304	-0.009700	-0.042920	-0.025988	...	-0.045309	0.049151	-0.049057	0.017821	-0.018061	-0.010441	0.043070	0.018440	-0.008274
I am feeling terrible	0.017495	-0.057904	0.033315	0.001710	0.051957	-0.048159	0.007659	0.119096	0.029929	-0.068960	...	0.038813	0.003015	-0.074585	-0.018391	-0.026449	0.005867	0.051495	-0.009829	0.030009

✓ 2s completed at 12:20

PYTHON + Google Colab



```
[6] import pandas as pd
    from sentence_transformers import SentenceTransformer
    from transformers import pipeline
```

Feature Extraction

We will begin by extracting features (numerical representations) from the text data using the `sentence-transformers` package. We will use the following three sentences, stored as a list of strings, as input to the model.:

```
[7] sentences = [
        "I feel great this morning",
        "I am feeling very good today",
        "I am feeling terrible"
    ]
```

We will use the `all-MiniLM-L6-v2` model to extract features from the sentences. The model will encode the sentences into a 384-dimensional vector representation. We will then print the features as a pandas dataframe for easy viewing.

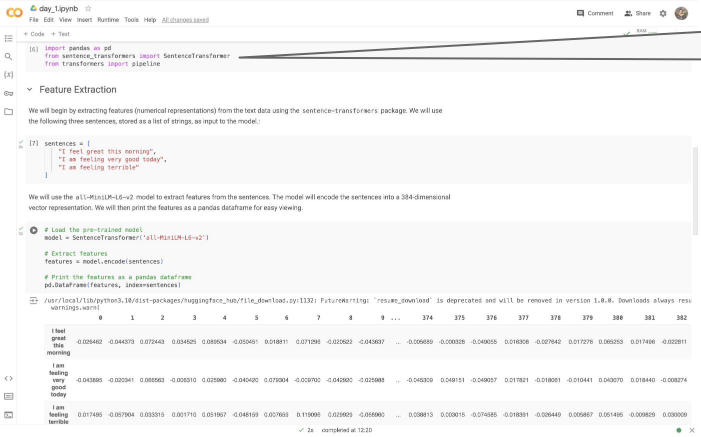
```
# Load the pre-trained model
model = SentenceTransformer('all-MiniLM-L6-v2')

# Extract features
features = model.encode(sentences)

# Print the features as a pandas dataframe
pd.DataFrame(features, index=sentences)
```

```
/usr/local/lib/python3.10/dist-packages/huggingface_hub/file_download.py:1132: FutureWarning: `resume_download` is deprecated and will be removed in version 1.0.0. Downloads always resu
  warnings.warn(
```

	0	1	2	3	4	5	6	7	8	9	...	374	375	376	377	378	379	380	381	382
I feel great this morning	-0.026462	-0.044373	0.072443	0.034525	0.089534	-0.050451	0.018811	0.071296	-0.020522	-0.043637	...	-0.005689	-0.000328	-0.049055	0.016308	-0.027642	0.017276	0.065253	0.017496	-0.022811
I am feeling very good today	-0.043895	-0.020341	0.066563	-0.006310	0.025980	-0.040420	0.079304	-0.009700	-0.042920	-0.025988	...	-0.045309	0.049151	-0.049057	0.017821	-0.018061	-0.010441	0.043070	0.018440	-0.008274
I am feeling terrible	0.017495	-0.057904	0.033315	0.001710	0.051957	-0.048159	0.007659	0.119096	0.029929	-0.068960	...	0.038813	0.003015	-0.074585	-0.018391	-0.026449	0.005867	0.051495	-0.009829	0.030009

✓ 2s completed at 12:20

PYTHON + Google Colab



```
[6] import pandas as pd
    from sentence_transformers import SentenceTransformer
    from transformers import pipeline
```

Feature Extraction

We will begin by extracting features (numerical representations) from the text data using the `sentence-transformers` package. We will use the following three sentences, stored as a list of strings, as input to the model.:

```
[7] sentences = [
        "I feel great this morning",
        "I am feeling very good today",
        "I am feeling terrible"
    ]
```

We will use the `all-MiniLM-L6-v2` model to extract features from the sentences. The model will encode the sentences into a 384-dimensional vector representation. We will then print the features as a pandas dataframe for easy viewing.

```
# Load the pre-trained model
model = SentenceTransformer('all-MiniLM-L6-v2')

# Extract features
features = model.encode(sentences)

# Print the features as a pandas dataframe
pd.DataFrame(features, index=sentences)
```

```
/usr/local/lib/python3.10/dist-packages/huggingface_hub/file_download.py:1132: FutureWarning: `resume_download` is deprecated and will be removed in version 1.0.0. Downloads always resu
  warnings.warn(
```

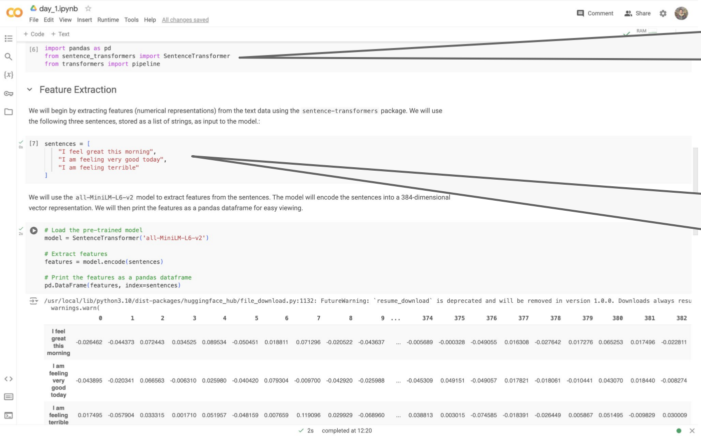
	0	1	2	3	4	5	6	7	8	9	...	374	375	376	377	378	379	380	381	382
I feel great this morning	-0.026462	-0.044373	0.072443	0.034525	0.089534	-0.050451	0.018811	0.071296	-0.020522	-0.043637	...	-0.005689	-0.000328	-0.049055	0.016308	-0.027642	0.017276	0.065253	0.017496	-0.022811
I am feeling very good today	-0.043895	-0.020341	0.066563	-0.006310	0.025980	-0.040420	0.079304	-0.009700	-0.042920	-0.025988	...	-0.045309	0.049151	-0.049057	0.017821	-0.018061	-0.010441	0.043070	0.018440	-0.008274
I am feeling terrible	0.017495	-0.057904	0.033315	0.001710	0.051957	-0.048159	0.007659	0.119096	0.029929	-0.068960	...	0.038813	0.003015	-0.074585	-0.018391	-0.026449	0.005867	0.051495	-0.009829	0.030009

✓ 2s completed at 12:20

package imports

```
import pandas as pd
from sentence_transformers import SentenceTransformer
from transformers import pipeline
```

PYTHON + Google Colab



```
[6] import pandas as pd
    from sentence_transformers import SentenceTransformer
    from transformers import pipeline
```

Feature Extraction

We will begin by extracting features (numerical representations) from the text data using the `sentence-transformers` package. We will use the following three sentences, stored as a list of strings, as input to the model.:

```
[7] sentences = [
        "I feel great this morning",
        "I am feeling very good today",
        "I am feeling terrible"
    ]
```

We will use the `all-MiniLM-L6-v2` model to extract features from the sentences. The model will encode the sentences into a 384-dimensional vector representation. We will then print the features as a pandas dataframe for easy viewing.

```
# Load the pre-trained model
model = SentenceTransformer('all-MiniLM-L6-v2')

# Extract features
features = model.encode(sentences)

# Print the features as a pandas dataframe
pd.DataFrame(features, index=sentences)
```

```
/usr/local/lib/python3.10/dist-packages/huggingface_hub/file_download.py:1132: FutureWarning: `resume_download` is deprecated and will be removed in version 1.0.0. Downloads always resu
  warnings.warn(
```

	0	1	2	3	4	5	6	7	8	9	...	374	375	376	377	378	379	380	381	382
I feel great this morning	-0.026462	-0.044373	0.072443	0.034525	0.089534	-0.050451	0.018811	0.071296	-0.020522	-0.043637	...	-0.005689	-0.000328	-0.049055	0.016308	-0.027642	0.017276	0.065253	0.017496	-0.022811
I am feeling very good today	-0.043895	-0.020341	0.066563	-0.006310	0.025980	-0.040420	0.079304	-0.009700	-0.042920	-0.025988	...	-0.045309	0.049151	-0.049057	0.017821	-0.018061	-0.010441	0.043070	0.018440	-0.008274
I am feeling terrible	0.017495	-0.057904	0.033315	0.001710	0.051957	-0.048159	0.007659	0.119096	0.029929	-0.068960	...	0.038813	0.003015	-0.074585	-0.018391	-0.026449	0.005867	0.051495	-0.009829	0.030009

✓ 2s completed at 12:20

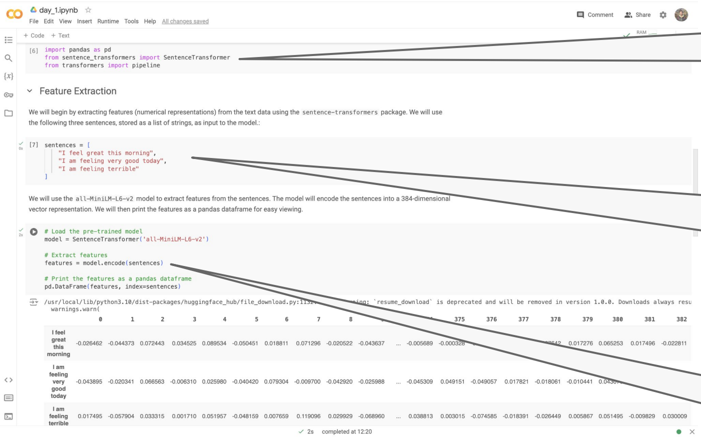
package imports

```
import pandas as pd
from sentence_transformers import SentenceTransformer
from transformers import pipeline
```

variable assignment, lists, strings

```
sentences = [
    "I feel great this morning",
    "I am feeling very good today",
    "I am feeling terrible"
]
```

PYTHON + Google Colab



```
[6] import pandas as pd
    from sentence_transformers import SentenceTransformer
    from transformers import pipeline
```

Feature Extraction

We will begin by extracting features (numerical representations) from the text data using the `sentence-transformers` package. We will use the following three sentences, stored as a list of strings, as input to the model.:

```
[7] sentences = [
        "I feel great this morning",
        "I am feeling very good today",
        "I am feeling terrible"
    ]
```

We will use the `all-MiniLM-L6-v2` model to extract features from the sentences. The model will encode the sentences into a 384-dimensional vector representation. We will then print the features as a pandas dataframe for easy viewing.

```
# Load the pre-trained model
model = SentenceTransformer('all-MiniLM-L6-v2')

# Extract features
features = model.encode(sentences)

# Print the features as a pandas dataframe
pd.DataFrame(features, index=sentences)
```

```
/usr/local/lib/python3.10/dist-packages/huggingface_hub/file_download.py:113[illegible] [illegible]ning: `resume_download` is deprecated and will be removed in version 1.0.0. Downloads always resu
  warnings.warn(
```

	0	1	2	3	4	5	6	7	8	[illegible]	...	[illegible]	375	376	377	378	379	380	381	382
I feel great this morning	-0.026462	-0.044373	0.072443	0.034525	0.089534	-0.050451	0.018811	0.071296	-0.020522	-0.043637	...	-0.005689	-0.000328	[illegible]	[illegible]	[illegible]	0.017276	0.065253	0.017496	-0.022811
I am feeling very good today	-0.043895	-0.020341	0.066563	-0.006310	0.025980	-0.040420	0.079304	-0.009700	-0.042920	-0.025988	...	-0.045309	0.049151	-0.049057	0.017821	-0.018061	-0.010441	[illegible]	[illegible]	[illegible]
I am feeling terrible	0.017495	-0.057904	0.033315	0.001710	0.051957	-0.048159	0.007659	0.119096	0.029929	-0.068960	...	0.038813	0.003015	-0.074585	-0.018391	-0.026449	0.005867	0.051495	-0.009829	0.030009

package imports

```
import pandas as pd
from sentence_transformers import SentenceTransformer
from transformers import pipeline
```

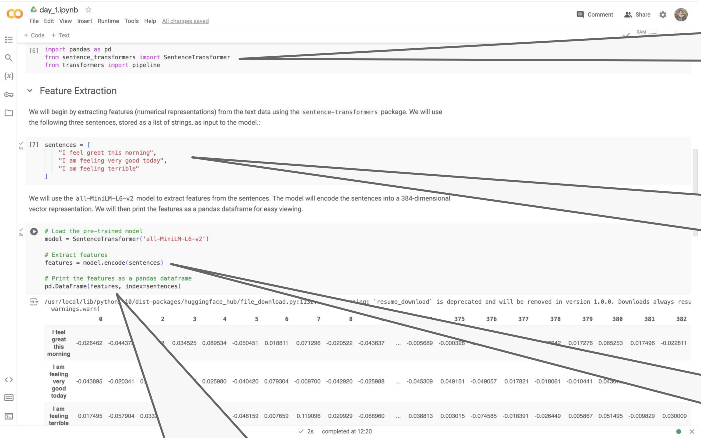
variable assignment, lists, strings

```
sentences = [
    "I feel great this morning",
    "I am feeling very good today",
    "I am feeling terrible"
]
```

dot notation, methods, attributes

```
# Extract features
features = model.encode(sentences)
```

PYTHON + Google Colab



```
[6] import pandas as pd
    from sentence_transformers import SentenceTransformer
    from transformers import pipeline
```

Feature Extraction

We will begin by extracting features (numerical representations) from the text data using the `sentence-transformers` package. We will use the following three sentences, stored as a list of strings, as input to the model.:

```
[7] sentences = [
        "I feel great this morning",
        "I am feeling very good today",
        "I am feeling terrible"
    ]
```

We will use the `all-MiniLM-L6-v2` model to extract features from the sentences. The model will encode the sentences into a 384-dimensional vector representation. We will then print the features as a pandas dataframe for easy viewing.

```
# Load the pre-trained model
model = SentenceTransformer('all-MiniLM-L6-v2')

# Extract features
features = model.encode(sentences)

# Print the features as a pandas dataframe
pd.DataFrame(features, index=sentences)
```

```
/usr/local/lib/python[illegible]10/dist-packages/huggingface_hub/file_download.py:1132: [illegible]ning: `resume_download` is deprecated and will be removed in version 1.0.0. Downloads always resu
  warnings.warn(
```

	0	[illegible]	2	3	4	5	6	7	8	[illegible]	[illegible]	[illegible]	375	376	377	378	379	380	381	382
I feel great this morning	-0.026462	-0.04437[illegible]	[illegible]	0.034525	0.089534	-0.050451	0.018811	0.071296	-0.020522	-0.043637	...	-0.005689	-0.000328	[illegible]	[illegible]	[illegible]	0.017276	0.065253	0.017496	-0.022811
I am feeling very good today	-0.043895	-0.020341	0.[illegible]	[illegible]	0.025980	-0.040420	0.079304	-0.009700	-0.042920	-0.025988	...	-0.045309	0.049151	-0.049057	0.017821	-0.018061	-0.010441	0.0430[illegible]	[illegible]	[illegible]
I am feeling terrible	0.017495	-0.057904	0.033[illegible]	[illegible]	[illegible]	-0.048159	0.007659	0.119096	0.029929	-0.068960	...	0.038813	0.003015	-0.074585	-0.018391	-0.026449	0.005867	0.051495	-0.009829	0.030009

package imports

```
import pandas as pd
from sentence_transformers import SentenceTransformer
from transformers import pipeline
```

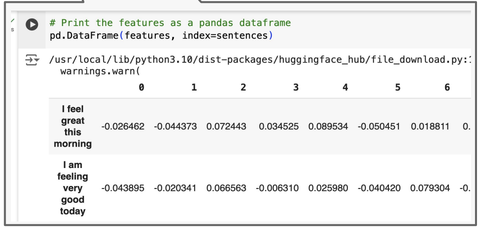
variable assignment, lists, strings

```
sentences = [
    "I feel great this morning",
    "I am feeling very good today",
    "I am feeling terrible"
]
```

dot notation, methods, attributes

```
# Extract features
features = model.encode(sentences)
```

printing



```
# Print the features as a pandas dataframe
pd.DataFrame(features, index=sentences)
```

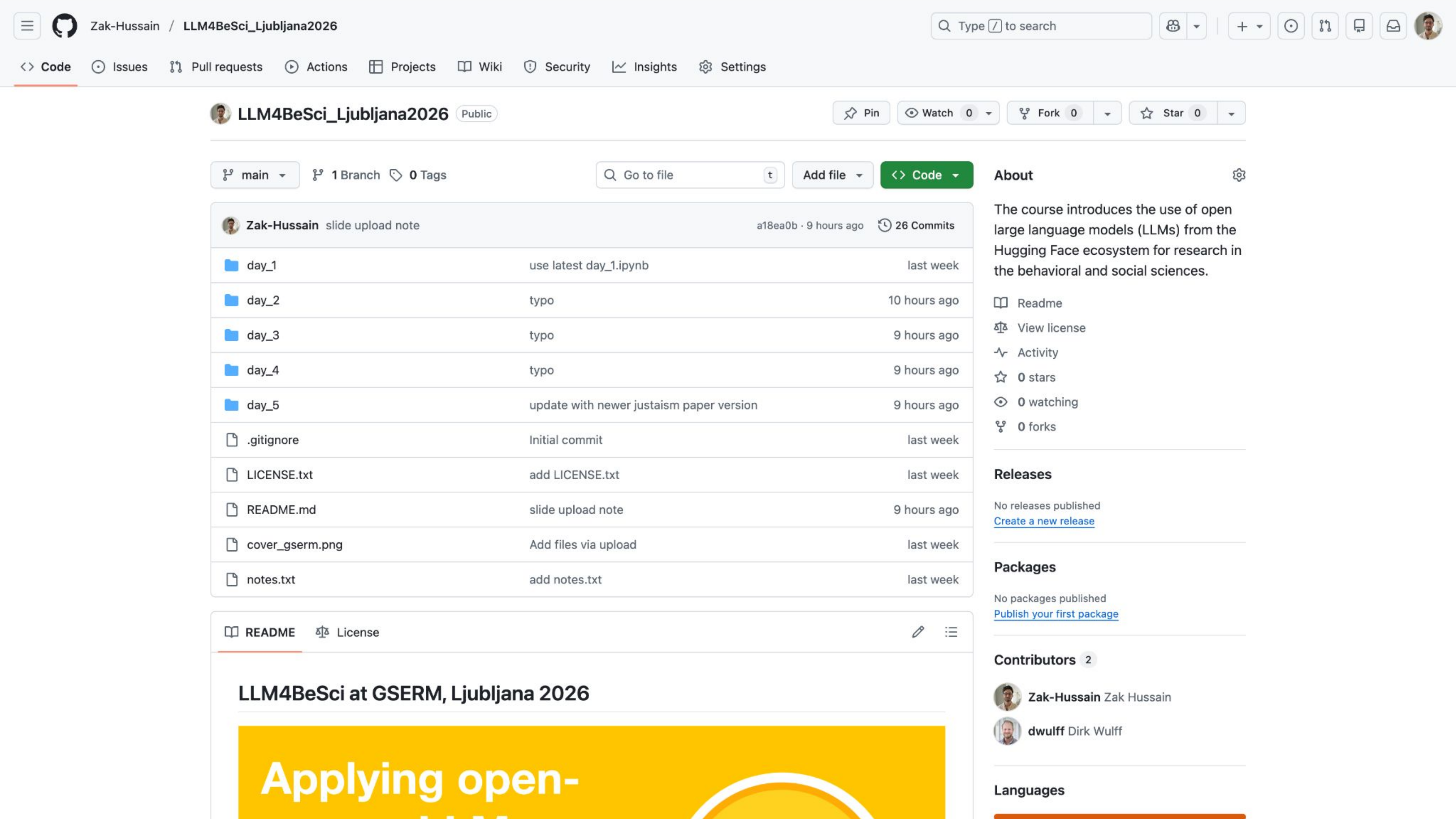
```
/usr/local/lib/python3.10/dist-packages/huggingface_hub/file_download.py:
  warnings.warn(
```

	0	1	2	3	4	5	6	
I feel great this morning	-0.026462	-0.044373	0.072443	0.034525	0.089534	-0.050451	0.018811	0.
I am feeling very good today	-0.043895	-0.020341	0.066563	-0.006310	0.025980	-0.040420	0.079304	-0.

OUR SOFTWARE STACK

python™ + 🤗 + Google colab

Time to install our stack…



LLM4BeSci_Ljubljana2026

Zak-Hussain slide upload note a18ea0b · 9 hours ago 26 Commits

day_1	use latest day_1.ipynb	last week
day_2	typo	10 hours ago
day_3	typo	9 hours ago
day_4	typo	9 hours ago
day_5	update with newer justaism paper version	9 hours ago
.gitignore	Initial commit	last week
LICENSE.txt	add LICENSE.txt	last week
README.md	slide upload note	9 hours ago
cover_gserm.png	Add files via upload	last week
notes.txt	add notes.txt	last week

LLM4BeSci at GSERM, Ljubljana 2026

About

The course introduces the use of open large language models (LLMs) from the Hugging Face ecosystem for research in the behavioral and social sciences.

- Readme
- View license
- Activity
- 0 stars
- 0 watching
- 0 forks

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Contributors 2

Zak-Hussain Zak Hussain
dwulff Dirk Wulff

Languages