

Start coding or [generate](#) with AI.

```
# NLP Pipelines with Hugging Face Transformers
# 🍷 NLP Pipelines with Transformers
```

This project showcases 8 major NLP tasks using Hugging Face's `transformers` library:

1. ✅ Text Classification
2. ❓ Zero-Shot Classification
3. 🗣️ Text Generation
4. 🧩 Masked Language Modeling
5. 📄 Named Entity Recognition (Token Classification)
6. 🔍 Question Answering
7. ✂️ Summarization
8. 🌐 Translation

```
## 🛠️ Setup
```

```
```bash
pip install transformers torch
```

[+ Code](#)
[+ Text](#)

```
from transformers import pipeline
```

```
'''1. Text Classification
```

```
Goal: Put the text into a category (positive/negative, spam/ham, etc.)
```

```
Used for: Sentiment analysis, spam detection, topic labeling'''
```

```
classifier = pipeline("sentiment-analysis")
```

```
classifier(["I've been waiting for a Hugging Face course my Whole life", "I hate this Generation so much", "My meeting on 5PM today"])
```

```
➡ No model was supplied, defaulted to distilbert/distilbert-base-uncased-finetuned-sst-2-english and revision 714eb0f (https://huggingface.co/distilbert/distilbert-base-uncased-finetuned-sst-2-english)
Using a pipeline without specifying a model name and revision in production is not recommended.
Device set to use cpu
[{'label': 'POSITIVE', 'score': 0.9982088804244995},
 {'label': 'NEGATIVE', 'score': 0.999135434627533},
 {'label': 'POSITIVE', 'score': 0.9751960039138794}]
```

```
'''2. Zero-Shot Classification
```

```
Goal: Classify text into categories without training the model on those categories.
```

```
Used for: Dynamic categorization when labels are unknown at training time.'''
```

```
classifier = pipeline("zero-shot-classification")
```

```
classifier("This is the course about the transformers library", candidate_labels = ["education", "Positive", "Business"] )
```

```
➡ No model was supplied, defaulted to facebook/bart-large-mnli and revision d7645e1 (https://huggingface.co/facebook/bart-large-mnli)
Using a pipeline without specifying a model name and revision in production is not recommended.
config.json:      1.15k/? [00:00<00:00, 77.5kB/s]
model.safetensors: 100%                               1.63G/1.63G [00:33<00:00, 30.5MB/s]
tokenizer_config.json: 100%                             26.0/26.0 [00:00<00:00, 2.07kB/s]
vocab.json:       899k/? [00:00<00:00, 25.7MB/s]
merges.txt:       456k/? [00:00<00:00, 21.2MB/s]
tokenizer.json:   1.36M/? [00:00<00:00, 43.4MB/s]
Device set to use cpu
{'sequence': 'This is the course about the transformers library',
 'labels': ['education', 'Positive', 'Business'],
 'scores': [0.8236019015312195, 0.14179572463035583, 0.034602370113134384]}
```

```
'''3.1 Text Generation
```

```
Goal: Generate new text based on a prompt.
```

```
Used for: Story writing, content generation, AI chatbots.'''
```

```
generator = pipeline("text-generation")
generator("The Generative AI is the Most Importat for")
```

➡ No model was supplied, defaulted to openai-community/gpt2 and revision 607a30d (<https://huggingface.co/openai-community/gpt2>). Using a pipeline without specifying a model name and revision in production is not recommended.

config.json:	100%	665/665	[00:00<00:00, 53.8kB/s]
model.safetensors:	100%	548M/548M	[00:30<00:00, 17.0MB/s]
generation_config.json:	100%	124/124	[00:00<00:00, 11.3kB/s]
tokenizer_config.json:	100%	26.0/26.0	[00:00<00:00, 2.41kB/s]
vocab.json:	1.04M/?	[00:00<00:00, 22.9MB/s]	
merges.txt:	456k/?	[00:00<00:00, 17.4MB/s]	
tokenizer.json:	1.36M/?	[00:00<00:00, 33.9MB/s]	

```
Device set to use cpu
Setting 'pad_token_id' to 'eos_token_id':50256 for open-end generation.
[{'generated_text': "The Generative AI is the Most Impactat for the Future?\n\nThe Generative AI is the most impactful and scalable solution for artificial intelligence. The Generative AI is an incredibly efficient, flexible, and cost-effective solution to many of the traditional problems associated with AI. It is a great way to think about how systems and systems of thought might interact in the future. It's also a great way to think about how AI could be used to solve our common problems.\n\nNow, this is going to sound a little crazy, but, in the world of AI and robotics, there's no better way to think of what computers are capable of doing than to think about how systems and systems of thought might interact in the future. For example, think about how you might think about how the world should be governed to the point where any form of technology that would be possible is achievable.\n\nWe've
```

```
# 3.2 Text Generation
# Goal: Generate new text based on a prompt.
# Used for: Story writing, content generation, AI chatbots.
```

```
from transformers import pipeline
```

```
# Create a text generation pipeline using the "distilgpt2" model
generator = pipeline("text-generation", model="distilgpt2")
```

```
# Generate text based on the provided prompt
# max_length: The maximum length of the generated text (including the prompt).
# num_return_sequences: The number of different sequences to generate.
generator("In this Gen-Z world, the generative AI play a Most Imortant Role is", max length=30, num return sequences=2)
```

```

→
nfig.json: 100%                                762/762 [00:00<00:00, 64.7kB/s]

odel.safetensors: 100%                        353M/353M [00:11<00:00, 38.5MB/s]

neration_config.json: 100%                    124/124 [00:00<00:00, 8.60kB/s]

kenizer_config.json: 100%                     26.0/26.0 [00:00<00:00, 2.68kB/s]

ocab.json: 100%                               1.04M/1.04M [00:00<00:00, 7.04MB/s]

erges.txt: 100%                               456k/456k [00:00<00:00, 10.7MB/s]

kenizer.json: 100%                           1.36M/1.36M [00:00<00:00, 13.0MB/s]

vice set to use cpu
uncation was not explicitly activated but `max_length` is provided a specific value, please use `truncation=True` to explicitly tr
tting `pad_token_id` to `eos_token_id`:50256 for open-end generation.
th `max_new_tokens` (=256) and `max_length` (=30) seem to have been set. `max_new_tokens` will take precedence. Please refer to the
`generated_text`: 'In this Gen-Z world, the generative AI play a Most Important Role is an example of the role of generative AI in
e evolution of the AI. From a theoretical perspective, the evolution of the AI is driven by a particular set of characteristics.
the first instance, the genetic code that controls the genetic code is modified by the individual genes. The evolution of the AI
driven by a particular set of characteristics. In the first instance, the genetic code that controls the genetic code is
modified by the individual genes. The evolution of the AI is driven by a particular set of characteristics. In the first instance,
the genetic code that controls the genetic code is modified by the individual genes. The evolution of the AI is driven by a
particular set of characteristics. In the first instance, the genetic code that controls the genetic code is modified by the
individual genes. The evolution of the AI is driven by a particular set of characteristics. In the first instance, the genetic code
that controls the genetic code is modified by the individual genes. The evolution of the AI is driven by a particular set of
characteristics. In the first instance, the genetic code that controls the genetic code is modified by the individual genes. The
evolution of the AI is driven by a particular set of characteristics. In the first instance, the genetic code that controls the
genetic code is modified by the',
`generated_text`: 'In this Gen-Z world, the generative AI play a Most Important Role is an AI that can be seen as the first AI to
e an AI and is used as a starting point in many other AI projects.\n\n\n\nThe main feature of the Gen-Z world is the ability to

```

'''4. Text Completion (Masked Language Modeling)  
Goal: Fill in the missing word in a sentence.

Used for: Auto-completion, grammar correction'''

```
unmasker = pipeline("fill-mask")
unmasker("The Gen AI teach you all about <mask> model", top k = 2)
```



⚡ No model was supplied, defaulted to distilbert/distilbert-base-cased-distilled-squad and revision 564e9b5 (<https://huggingface.co/d>). Using a pipeline without specifying a model name and revision in production is not recommended.

```
config.json: 100% 473/473 [00:00<00:00, 6.62kB/s]

model.safetensors: 100% 261M/261M [00:16<00:00, 16.4MB/s]

tokenizer_config.json: 100% 49.0/49.0 [00:00<00:00, 2.56kB/s]

vocab.txt: 213k/? [00:00<00:00, 13.8MB/s]

tokenizer.json: 436k/? [00:00<00:00, 22.5MB/s]

Device set to use cpu
{'score': 0.4327382743358612,
 'start': 39,
 'end': 72,
 'answer': 'Gen AI and Data Analytics Domain'}
```

'''7. Summarization

Goal: Create a short summary from a long text.

Used for: News summaries, article previews.'''

```
summarizer = pipeline("summarization")
summarizer("""Introduction In the modern landscape of business, organizations must constantly seek ways to harness technological advance
""", max_length=100, min_length=30)
```

⚡ No model was supplied, defaulted to sshleifer/distilbart-cnn-12-6 and revision a4f8f3e (<https://huggingface.co/sshleifer/distilbart>). Using a pipeline without specifying a model name and revision in production is not recommended.

```
config.json: 1.80k/? [00:00<00:00, 136kB/s]

pytorch_model.bin: 100% 1.22G/1.22G [00:23<00:00, 55.9MB/s]

model.safetensors: 60% 734M/1.22G [00:16<00:13, 35.6MB/s]

tokenizer_config.json: 100% 26.0/26.0 [00:00<00:00, 1.58kB/s]

vocab.json: 899k/? [00:00<00:00, 1.78MB/s]

merges.txt: 456k/? [00:00<00:00, 3.95MB/s]

Device set to use cpu
[{'summary_text': ' Gartner predicts that by 2026, more than 80% of businesses will use Generative AI technology . This is a
significant increase from less than 5% in 2023 . With just a few words of prompt, you can get answers in the form of text, image,
```

'''8. Translation

Goal: Translate from one language to another.

Used for: Multilingual apps, document conversion.'''

```
translator = pipeline("translation", model = "Helsinki-NLP/opus-mt-en-hi")
translator("I want to learn modern Technologies which help me to get job in this AI world")
```

⚡ Device set to use cpu  
[{'translation\_text': 'मैं इस एआई दुनिया में काम करने में मेरी मदद करना चाहते हैं कि आधुनिक टेक तकनीक के बारे में सीखना चाहते हैं'}]

Start coding or [generate](#) with AI.

# 1. Text Classification

```
def text_classification():
    classifier = pipeline("text-classification")
    text = "I absolutely love this phone!"
    result = classifier(text)
    print("1. Text Classification:\n", result, "\n")
```

# 2. Zero-Shot Classification

```
def zero_shot_classification():
    classifier = pipeline("zero-shot-classification")
    text = "This mobile has a long battery life and great camera."
    labels = ["technology", "sports", "politics"]
    result = classifier(text, candidate_labels=labels)
    print("2. Zero-Shot Classification:\n", result, "\n")
```

# 3. Text Generation

```
def text_generation():
    generator = pipeline("text-generation")
    prompt = "In the future, artificial intelligence will"
    result = generator(prompt, max_length=30, num_return_sequences=1)
    print("3. Text Generation:\n", result, "\n")
```

# 4. Masked Language Modeling

```
def masked_language_modeling():
    fill_mask = pipeline("fill-mask")
    masked_sentence = "Machine learning is the <mask> of AI."
    result = fill_mask(masked_sentence)
    print("4. Masked Language Modeling:\n", result, "\n")

# 5. Token Classification (NER)
def token_classification():
    ner = pipeline("token-classification")
    text = "Elon Musk founded SpaceX in the United States."
    result = ner(text)
    print("5. Token Classification (NER):\n", result, "\n")

# 6. Question Answering
def question_answering():
    qa = pipeline("question-answering")
    context = "Barack Obama was born in Hawaii. He was the 44th president of the USA."
    question = "Where was Obama born?"
    result = qa(question=question, context=context)
    print("6. Question Answering:\n", result, "\n")

# 7. Summarization
def summarization():
    summarizer = pipeline("summarization")
    long_text = ("Artificial intelligence and machine learning are rapidly changing the world. "
                 "They are being used in industries such as healthcare, finance, and transportation "
                 "to improve efficiency, reduce costs, and enhance decision-making.")
    result = summarizer(long_text, max_length=40, min_length=10, do_sample=False)
    print("7. Summarization:\n", result, "\n")

# 8. Translation
def translation():
    translator = pipeline("translation", model="Helsinki-NLP/opus-mt-en-hi")
    text = "I love learning natural language processing."
    result = translator(text)
    print("8. Translation (English to Hindi):\n", result, "\n")

# Run all
if __name__ == "__main__":
    text_classification()
    zero_shot_classification()
    text_generation()
    masked_language_modeling()
    token_classification()
    question_answering()
    summarization()
    translation()
```

➡ No model was supplied, defaulted to distilbert/distilbert-base-uncased-finetuned-sst-2-english and revision 714eb0f (<https://huggingface.co/distilbert/distilbert-base-uncased-finetuned-sst-2-english>)  
Using a pipeline without specifying a model name and revision in production is not recommended.

Device set to use cpu

No model was supplied, defaulted to facebook/bart-large-mnli and revision d7645e1 (<https://huggingface.co/facebook/bart-large-mnli>)  
Using a pipeline without specifying a model name and revision in production is not recommended.

1. Text Classification:

```
[{'label': 'POSITIVE', 'score': 0.9998652935028076}]
```

Device set to use cpu

No model was supplied, defaulted to openai-community/gpt2 and revision 607a30d (<https://huggingface.co/openai-community/gpt2>).  
Using a pipeline without specifying a model name and revision in production is not recommended.

2. Zero-Shot Classification:

```
{ 'sequence': 'This mobile has a long battery life and great camera.', 'labels': ['technology', 'sports', 'politics'], 'scores': [
```

Device set to use cpu

Truncation was not explicitly activated but `max\_length` is provided a specific value, please use `truncation=True` to explicitly  
Setting `pad\_token\_id` to `eos\_token\_id`:50256 for open-end generation.  
Both `max\_new\_tokens` (=256) and `max\_length` (=30) seem to have been set. `max\_new\_tokens` will take precedence. Please refer to t  
No model was supplied, defaulted to distilbert/distilroberta-base and revision fb53ab8 (<https://huggingface.co/distilbert/distilroberta-base>)  
Using a pipeline without specifying a model name and revision in production is not recommended.

3. Text Generation:

```
[{'generated_text': "In the future, artificial intelligence will be able to predict the future by analyzing information that peop
```

Some weights of the model checkpoint at distilbert/distilroberta-base were not used when initializing RobertaForMaskedLM: ['robert  
- This IS expected if you are initializing RobertaForMaskedLM from the checkpoint of a model trained on another task or with anoth  
- This IS NOT expected if you are initializing RobertaForMaskedLM from the checkpoint of a model that you expect to be exactly ide  
Device set to use cpu  
/usr/local/lib/python3.11/dist-packages/torch/nn/modules/module.py:1750: FutureWarning: `encoder\_attention\_mask` is deprecated and  
return forward\_call(\*args, \*\*kwargs)  
No model was supplied, defaulted to dbmdz/bert-large-cased-finetuned-conll03-english and revision 4c53496 (<https://huggingface.co/dbmdz/bert-large-cased-finetuned-conll03-english>)  
Using a pipeline without specifying a model name and revision in production is not recommended.

4. Masked Language Modeling:

```
[{'score': 0.3512125313282013, 'token': 499, 'token_str': ' future', 'sequence': 'Machine learning is the future of AI.'}, {'scor
```

Some weights of the model checkpoint at dbmdz/bert-large-cased-finetuned-conll03-english were not used when initializing BertForTo  
- This IS expected if you are initializing BertForTokenClassification from the checkpoint of a model trained on another task or wi  
- This IS NOT expected if you are initializing BertForTokenClassification from the checkpoint of a model that you expect to be exa  
Device set to use cpu  
/usr/local/lib/python3.11/dist-packages/torch/nn/modules/module.py:1750: FutureWarning: `encoder\_attention\_mask` is deprecated and  
return forward\_call(\*args, \*\*kwargs)  
No model was supplied, defaulted to distilbert/distilbert-base-cased-distilled-squad and revision 564e9b5 (<https://huggingface.co/distilbert/distilbert-base-cased-distilled-squad>)  
Using a pipeline without specifying a model name and revision in production is not recommended.

5. Token Classification (NER):

```
[{'entity': 'I-PER', 'score': np.float32(0.9994566), 'index': 1, 'word': 'El', 'start': 0, 'end': 2}, {'entity': 'I-PER', 'score'
```

Device set to use cpu

No model was supplied, defaulted to sshleifer/distilbart-cnn-12-6 and revision a4f8f3e (<https://huggingface.co/sshleifer/distilbart-cnn-12-6>)  
Using a pipeline without specifying a model name and revision in production is not recommended.

6. Question Answering:

```
{ 'score': 0.9848431944847107, 'start': 25, 'end': 31, 'answer': 'Hawaii'}
```

Device set to use cpu