```
!pip install transformers sentencepiece
```

```
Requirement already satisfied: transformers in /usr/local/lib/python3.11/dist-packages (4.52.4)
Requirement already satisfied: sentencepiece in /usr/local/lib/python3.11/dist-packages (0.2.0)
Requirement already satisfied: filelock in /usr/local/lib/python3.11/dist-packages (from transformers) (3.18.0)
Requirement already satisfied: huggingface-hub<1.0,>=0.30.0 in /usr/local/lib/python3.11/dist-packages (from transformers) (0.32.4)
Requirement already satisfied: numpy>=1.17 in /usr/local/lib/python3.11/dist-packages (from transformers) (2.0.2)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.11/dist-packages (from transformers) (24.2)
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.11/dist-packages (from transformers) (6.0.2)
Requirement already satisfied: regex!=2019.12.17 in /usr/local/lib/python3.11/dist-packages (from transformers) (2024.11.6)
Requirement already satisfied: requests in /usr/local/lib/python3.11/dist-packages (from transformers) (2.32.3)
Requirement already satisfied: tokenizers<0.22,>=0.21 in /usr/local/lib/python3.11/dist-packages (from transformers) (0.21.1)
Requirement already satisfied: safetensors>=0.4.3 in /usr/local/lib/python3.11/dist-packages (from transformers) (0.5.3)
Requirement already satisfied: tqdm>=4.27 in /usr/local/lib/python3.11/dist-packages (from transformers) (4.67.1)
Requirement already satisfied: fsspec>=2023.5.0 in /usr/local/lib/python3.11/dist-packages (from huggingface-hub<1.0,>=0.30.0->transform
Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/lib/python3.11/dist-packages (from huggingface-hub<1.0,>=0.30.0-
Requirement already satisfied: hf-xet<2.0.0,>=1.1.2 in /usr/local/lib/python3.11/dist-packages (from huggingface-hub<1.0,>=0.30.0->trans
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.11/dist-packages (from requests->transformers) (3.4.2)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.11/dist-packages (from requests->transformers) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.11/dist-packages (from requests->transformers) (2.4.0)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.11/dist-packages (from requests->transformers) (2025.4.26)
```

```
from transformers import AutoTokenizer, AutoModelForSeq2SeqLM

model_name = "facebook/nllb-200-distilled-600M"
tokenizer = AutoTokenizer.from_pretrained(model_name)
model = AutoModelForSeq2SeqLM.from_pretrained(model_name)
```

```
/usr/local/lib/python3.11/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), set it as secre
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(
tokenizer_config.json:  100%                         564/564  [00:00<00:00,  12.6kB/s]

sentencepiece.bpe.model:  100%                       4.85M/4.85M  [00:00<00:00,  7.45MB/s]

tokenizer.json:  100%                                17.3M/17.3M  [00:00<00:00,  20.8MB/s]

special_tokens_map.json:  100%                       3.55k/3.55k  [00:00<00:00,  109kB/s]

config.json:  100%                                   846/846  [00:00<00:00,  24.4kB/s]

pytorch_model.bin:  100%                             2.46G/2.46G  [00:54<00:00,  28.9MB/s]

model.safetensors:  100%                             2.46G/2.46G  [02:32<00:00,  39.5MB/s]

generation_config.json:  100%                        189/189  [00:00<00:00,  4.76kB/s]
```

```
language_codes = {
    "Kannada": "kan_Knda",
    "French": "fra_Latn",
    "Spanish": "spa_Latn",
    "Hindi": "hin_Deva"
}
```

```
text = input("Enter your text here: ")
print("Original Text:", text)
```

```
Enter your text here: Hi this is Nikhil and i love travelling
Original Text: Hi this is Nikhil and i love travelling
```

```
print("\nTranslated Versions:")
for language, code in language_codes.items():
    tokenizer.src_lang = "eng_Latn"  # Source language: English
    encoded = tokenizer(text, return_tensors="pt")
    bos_token_id = tokenizer.convert_tokens_to_ids(code)

    generated_tokens = model.generate(
        **encoded,
        forced_bos_token_id=bos_token_id,
        max_length=50
    )
    translated = tokenizer.batch_decode(generated_tokens, skip_special_tokens=True)[0]
    print(f"{language}: {translated}")
```

```
Translated Versions:
Kannada: ಹಲೋ , ನಾನು ನಿಖಿಲ್ ಮತ್ತು ನಾನು ಪ್ರಯಾಣ ಮಾಡಲು ಇಷ್ಟಪಡುತ್ತೇನೆ
French: Bonjour , je suis Nikhil et j' aime voyager .
Spanish: Hola, soy Nikhil y me encanta viajar.
Hindi: हाय यह Nikhil है और मुझे यात्रा करना पसंद है
```

Start coding or generate with AI.