

## 1. Mount to Google Drive

```
from google.colab import drive
drive.mount('/content/drive')
```

Mounted at /content/drive

## 2. Install Libraries

```
!pip install -q transformers datasets rouge-score accelerate
!pip install -q sentencepiece
import pandas as pd
from transformers import T5Tokenizer
from datasets import Dataset
```

Preparing metadata (setup.py) ... done

=====	363.4/363.4 MB	4.1 MB/s	eta 0:00:00
=====	13.8/13.8 MB	36.5 MB/s	eta 0:00:00
=====	24.6/24.6 MB	44.8 MB/s	eta 0:00:00
=====	883.7/883.7 kB	32.7 MB/s	eta 0:00:00
=====	664.8/664.8 MB	1.2 MB/s	eta 0:00:00
=====	211.5/211.5 MB	2.9 MB/s	eta 0:00:00
=====	56.3/56.3 MB	11.7 MB/s	eta 0:00:00
=====	127.9/127.9 MB	9.3 MB/s	eta 0:00:00
=====	207.5/207.5 MB	5.4 MB/s	eta 0:00:00
=====	21.1/21.1 MB	27.6 MB/s	eta 0:00:00

Building wheel for rouge-score (setup.py) ... done

## 3. Load the CSV Files that Uploaded on Google Drive

```
train_df = pd.read_csv('/content/drive/MyDrive/cnn_dailymail/train.csv')
val_df = pd.read_csv('/content/drive/MyDrive/cnn_dailymail/validation.csv')
test_df = pd.read_csv('/content/drive/MyDrive/cnn_dailymail/test.csv')
```

```
print(train_df.columns)
train_df.head()
```

Index(['id', 'article', 'highlights'], dtype='object')

	id	article	highlights
0	0001d1afc246a7964130f43ae940af6bc6c57f01	By . Associated Press . PUBLISHED: . 14:11 EST...	Bishop John Folda, of North Dakota, is taking ...
1	0002095e55fcbd3a2f366d9bf92a95433dc305ef	(CNN) -- Ralph Mata was an internal affairs li...	Criminal complaint: Cop used his role to help ...
2	00027e965c8264c35cc1bc55556db388da82b07f	A drunk driver who killed a young woman in a h...	Craig Eccleston-Todd, 27, had drunk at least t...
3	0002c17436637c4fe1837c935c04de47adb18e9a	(CNN) -- With a breezy sweep of his pen Presid...	Nina dos Santos says Europe must be ready to a...
4	0003ad6ef0c37534f80b55b4235108024b407f0b	Fleetwood are the only team still to have a 10...	Fleetwood top of League One after 2-0 win at S...

```
# Sample smaller datasets for faster training
train_df = train_df.sample(5000, random_state=42).reset_index(drop=True)
val_df = val_df.sample(1000, random_state=42).reset_index(drop=True)
test_df = test_df.sample(1000, random_state=42).reset_index(drop=True)
```

## 4. Data Preprocessing


```
train_df = train_df[['article', 'highlights']].dropna()
val_df = val_df[['article', 'highlights']].dropna()
test_df = test_df[['article', 'highlights']].dropna()
```

```
# Rename for model compatibility
train_df.columns = ['text', 'summary']
val_df.columns = ['text', 'summary']
test_df.columns = ['text', 'summary']
```

## 5. Hugging Face Tokenizer

```
model_name = "t5-small"
tokenizer = T5Tokenizer.from_pretrained(model_name)
```

```
def tokenize_function(batch):
    input_encodings = tokenizer(batch['text'], truncation=True, padding='max_length', max_length=512)
    target_encodings = tokenizer(batch['summary'], truncation=True, padding='max_length', max_length=150)
    input_encodings['labels'] = target_encodings['input_ids']
    return input_encodings
```

 /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 secret.  
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% 2.32k/2.32k [00:00<00:00, 114kB/s]  
spiece.model: 100% 792k/792k [00:00<00:00, 9.60MB/s]  
tokenizer.json: 100% 1.39M/1.39M [00:00<00:00, 13.0MB/s]  
You are using the default legacy behaviour of the <class 'transformers.models.t5.tokenization\_t5.T5Tokenizer'>. This is expected, and si

## 6. Convert DataFrames to HuggingFace Format

```
train_dataset = Dataset.from_pandas(train_df)
val_dataset = Dataset.from_pandas(val_df)
test_dataset = Dataset.from_pandas(test_df)
```


```
train_dataset = train_dataset.map(tokenize_function, batched=True)
val_dataset = val_dataset.map(tokenize_function, batched=True)
test_dataset = test_dataset.map(tokenize_function, batched=True)
```

```
# Set format for PyTorch
train_dataset.set_format(type='torch', columns=['input_ids', 'attention_mask', 'labels'])
val_dataset.set_format(type='torch', columns=['input_ids', 'attention_mask', 'labels'])
test_dataset.set_format(type='torch', columns=['input_ids', 'attention_mask', 'labels'])
```

 Map: 100% 5000/5000 [01:20<00:00, 73.38 examples/s]  
Map: 100% 1000/1000 [00:09<00:00, 103.69 examples/s]  
Map: 100% 1000/1000 [00:08<00:00, 115.78 examples/s]


## 7. Save the Preprocessed Dataset in Disk

```
train_dataset.save_to_disk("/content/train_dataset")
val_dataset.save_to_disk("/content/val_dataset")
test_dataset.save_to_disk("/content/test_dataset")
```

 Saving the dataset (1/1 shards): 100% 5000/5000 [00:00<00:00, 74099.34 examples/s]  
Saving the dataset (1/1 shards): 100% 1000/1000 [00:00<00:00, 30223.34 examples/s]  
Saving the dataset (1/1 shards): 100% 1000/1000 [00:00<00:00, 27629.19 examples/s]

## 8. Fix NumPy Compatibility Issue with Hugging Face Datasets

```
!pip uninstall -y thinc spacy
!pip install -q --force-reinstall numpy==1.26.4
import os
os.kill(os.getpid(), 9)
```

 Found existing installation: thinc 8.3.6  
Uninstalling thinc-8.3.6:  
Successfully uninstalled thinc-8.3.6  
Found existing installation: spacy 3.8.6  
Uninstalling spacy-3.8.6:  
Successfully uninstalled spacy-3.8.6  
61.0/61.0 kB 2.5 MB/s eta 0:00:00  
18.3/18.3 MB 93.5 MB/s eta 0:00:00

## 9. Read the Saved Dataset

```
from datasets import load_from_disk

train_dataset = load_from_disk("/content/train_dataset")
val_dataset = load_from_disk("/content/val_dataset")
test_dataset = load_from_disk("/content/test_dataset")
```

## 10. Import Libraries

```
from transformers import T5ForConditionalGeneration, Trainer, TrainingArguments
from datasets import load_metric
import numpy as np
```

## 11. Load T5-small Model and Train


```
model_name = "t5-small"

model = T5ForConditionalGeneration.from_pretrained(model_name)
```

```
training_args = TrainingArguments(
    output_dir="./results",
    eval_strategy="epoch",
    save_strategy="epoch",
    per_device_train_batch_size=4,
    per_device_eval_batch_size=4,
    num_train_epochs=1,
    weight_decay=0.01,
    logging_dir="./logs",
    logging_steps=10,
    push_to_hub=False,
    load_best_model_at_end=True,
    report_to="none"
)
```

```
trainer = Trainer(
    model=model,
    args=training_args,
    train_dataset=train_dataset,
    eval_dataset=val_dataset,
)
```

```
trainer.train()
```

 /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 secret.  
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(
```

```
config.json: 100% 1.21k/1.21k [00:00<00:00, 27.5kB/s]
```

Xet Storage is enabled for this repo, but the 'hf\_xet' package is not installed. Falling back to regular HTTP download. For better performance, install the 'hf\_xet' package.

```
WARNING:huggingface_hub.file_download:Xet Storage is enabled for this repo, but the 'hf_xet' package is not installed. Falling back to regular HTTP download. For better performance, install the 'hf_xet' package.
model.safetensors: 100% 242M/242M [00:02<00:00, 115MB/s]
```

```
generation_config.json: 100% 147/147 [00:00<00:00, 2.59kB/s]
```

Passing a tuple of `past\_key\_values` is deprecated and will be removed in Transformers v4.48.0. You should pass an instance of `EncoderDecoderCache` instead.

Epoch	Training Loss	Validation Loss
1	1.154400	1.027472

There were missing keys in the checkpoint model loaded: ['encoder.embed\_tokens.weight', 'decoder.embed\_tokens.weight', 'lm\_head.weight']  
TrainOutput(global\_step=1250, training\_loss=1.259596794128418, metrics={'train\_runtime': 16345.9291, 'train\_samples\_per\_second': 0.306, 'train\_steps\_per\_second': 0.076, 'total\_flos': 67670900736000.0, 'train\_loss': 1.259596794128418, 'epoch': 1.0})

## 12. Evaluate the Model with ROUGE Score

```
!pip install evaluate
```


```
from evaluate import load
rouge = load("rouge")
eval_results = trainer.evaluate()
```

```
print("Evaluation Results (ROUGE):")
for key, value in eval_results.items():
    print(f"{key}: {value:.2f}")
```

 [250/250 15:00]

```
Evaluation Results (ROUGE):
eval_loss: 1.03
eval_runtime: 910.89
eval_samples_per_second: 1.10
eval_steps_per_second: 0.27
```

```
print(test_dataset.column_names)
```

 ['text', 'summary', 'input\_ids', 'attention\_mask', 'labels']

```
from transformers import T5Tokenizer
```

```
# tokenizer
tokenizer = T5Tokenizer.from_pretrained("t5-small")
```

```
for i in range(3):
    input_ids = test_dataset[i]["input_ids"]
    input_text = tokenizer.decode(input_ids, skip_special_tokens=True)
    summary = summarize(input_text)
    print(f"\nOriginal Text:\n{input_text[:500]}...\n")
    print(f"Generated Summary:\n{summary}\n")
```

 You are using the default legacy behaviour of the <class 'transformers.models.t5.tokenization\_t5.T5Tokenizer'>. This is expected, and si

```
Original Text:
Comedian Jenny Eclair travelled with her other half on a Painting In Venus break with Flavours. There comes a time in a woman's life whe
```

```
Generated Summary:
Jenny Eclair travelled with her other half on a Painting In Venus break with Flavours. It reminds you how much weight you forgot to lose
```

```
Original Text:
A woman of Arab and Jewish descent who was strip-searched at a Detroit-area airport has reached a settlement in a lawsuit filed on her b
```

```
Generated Summary:
Shoshana Hebshi, of Sylvania, Ohio, was strip-searched at Detroit Metropolitan Airport on the 10th anniversary of 9/11. The federal gove
```

```
Original Text:
World No 1 Novak Djokovic has apologised to the startled ball boy caught in the crossfire of a tirade at his support team during his win
```

```
Generated Summary:
Novak Djokovic snatched a towel from a ball boy caught in the crossfire. During the rant, Djokovic snatched a towel from the youngster.
```

### 13. Test on New Data

```
preds = []
refs = []

for example in test_dataset:
    # Decode input_ids to text
    input_text = tokenizer.decode(example["input_ids"], skip_special_tokens=True)
    label_ids = example["labels"]
    label_ids = [token if token != -100 else tokenizer.pad_token_id for token in label_ids]
    reference = tokenizer.decode(label_ids, skip_special_tokens=True)

    generated = summarize(input_text)

    preds.append(generated)
    refs.append(reference)

# Compute ROUGE
test_rouge = rouge.compute(predictions=preds, references=refs, use_stemmer=True)
```

```
print("Test ROUGE Scores:")
for key, value in test_rouge.items():
    print(f"{key}: {value * 100:.2f}")
```

→ Test ROUGE Scores:

rouge1:	38.45
rouge2:	17.76
rougeL:	27.52
rougeLsum:	27.53

#### 14. Save Model and Tokenizer

```
#model.save_pretrained("t5_summarizer_model")
#tokenizer.save_pretrained("t5_summarizer_model")
```