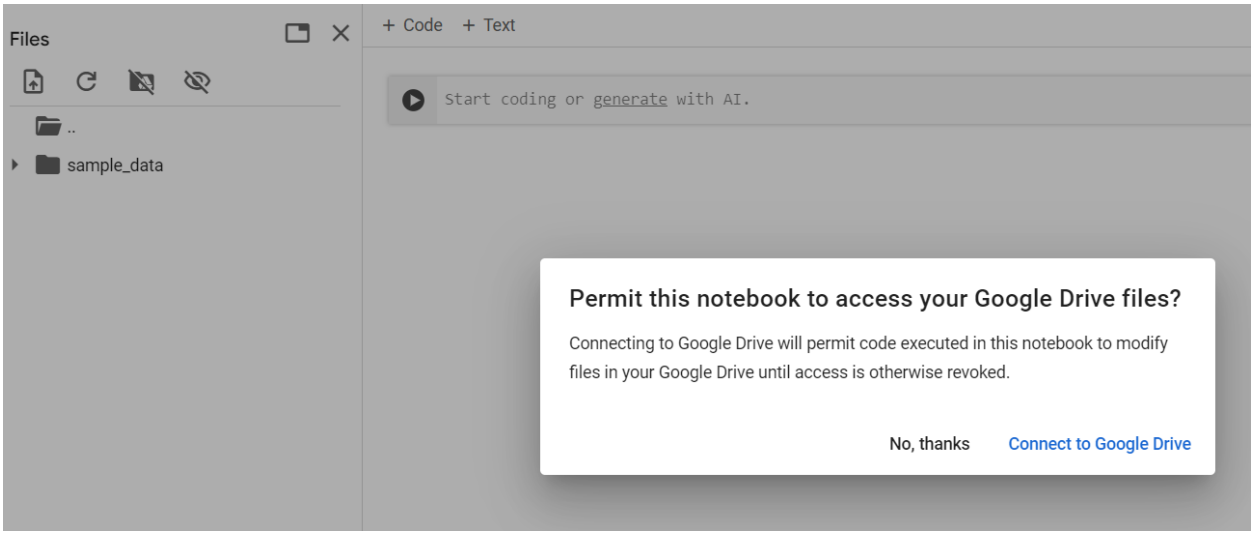
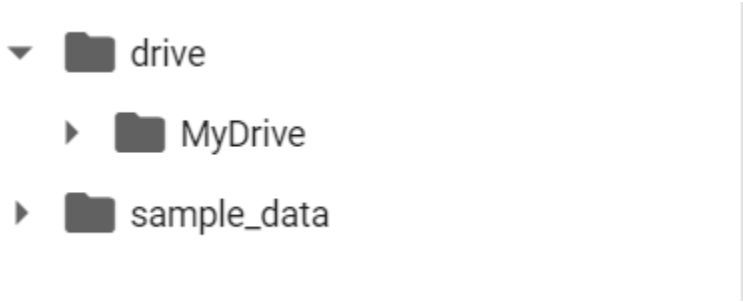
Team-MakeChange

Section – 2 Screen Captures:





```
!pip install transformers datasets pandas
```

```
  Downloading datasets-2.21.0-py3-none-any.whl.metadata (21 kB)
Requirement already satisfied: pandas in /usr/local/lib/python3.10/dist-packages (2.1.4)
Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-packages (from transformers) (3.
Requirement already satisfied: huggingface-hub<1.0,>=0.23.2 in /usr/local/lib/python3.10/dist-packages (fr
Requirement already satisfied: numpy>=1.17 in /usr/local/lib/python3.10/dist-packages (from transformers) (
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.10/dist-packages (from transformer
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.10/dist-packages (from transformers) (
Requirement already satisfied: regex!=2019.12.17 in /usr/local/lib/python3.10/dist-packages (from transfor
Requirement already satisfied: requests in /usr/local/lib/python3.10/dist-packages (from transformers) (2.3
Requirement already satisfied: safetensors>=0.4.1 in /usr/local/lib/python3.10/dist-packages (from transfor
Requirement already satisfied: tokenizers<0.20,>=0.19 in /usr/local/lib/python3.10/dist-packages (from tra
Requirement already satisfied: tqdm>=4.27 in /usr/local/lib/python3.10/dist-packages (from transformers) (4
Collecting pyarrow>=15.0.0 (from datasets)
  Downloading pyarrow-17.0.0-cp310-cp310-manylinux_2_28_x86_64.whl.metadata (3.3 kB)
Collecting dill<0.3.9,>=0.3.0 (from datasets)
  Downloading dill-0.3.8-py3-none-any.whl.metadata (10 kB)
Collecting xxhash (from datasets)
  Downloading xxhash-3.5.0-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (12 kB)
Collecting multiprocess (from datasets)
  Downloading multiprocess-0.70.16-py310-none-any.whl.metadata (7.2 kB)
Requirement already satisfied: fsspec<=2024.6.1,>=2023.1.0 in /usr/local/lib/python3.10/dist-packages (fro
Requirement already satisfied: aiohttp in /usr/local/lib/python3.10/dist-packages (from datasets) (3.10.5)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.10/dist-packages (from pan
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-packages (from pandas) (2024.
```

import lib

```
import pandas as pd
import matplotlib.pyplot as plt
from transformers import LEDTokenizer, LEDForConditionalGeneration
```

load csv

```
file_path = '/content/drive/MyDrive/wikiHow.csv'
df = pd.read_csv(file_path)
```

Display first few rows of the dataset

```
df.head()
```

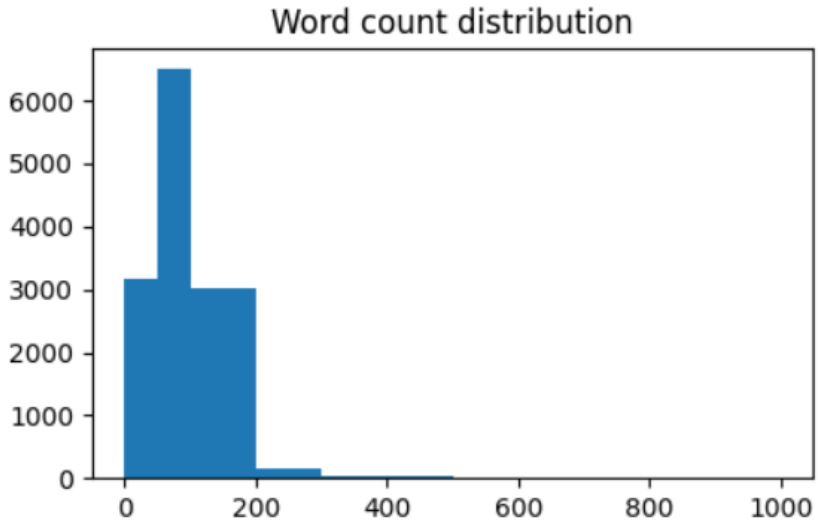
	Article Title	Subheading	
0	How to Rent a Post Office Box: 12 Steps (with ...	Fill out the application forms.	You can apply o
1	How to Rent a Post Office Box: 12 Steps (with ...	Prepare two forms of ID.	Whether you ap
2	How to Rent a Post Office Box: 12 Steps (with ...	Pay your fees in advance.	You can reserve
3	How to Rent a Post Office Box: 12 Steps (with ...	Collect your post office box keys.	You should receiv
4	How to Rent a Post Office Box: 12 Steps (with ...	Aim to collect your mail in a timely manner.	Given that ther

```
# Create the plot with a specified figure size
fig = plt.figure(figsize=(5, 3))

# Plot a histogram of the word count distribution, specifying the bin ranges
plt.hist(numOfWords.to_numpy(), bins=[0, 50, 100, 200, 300, 500, 1000])

# Add a title to the plot
plt.title("Word count distribution")

# Show the plot
plt.show()
```



nearly **200** outliers were removed

```
tempdf = df[df.length <= 200]
print(tempdf.shape)
```

```
(12686, 4)
```

```
#import model version
tokenizer = LEDTokenizer.from_pretrained("allenai/led-base-16384")
```

```
/usr/local/lib/python3.10/dist-packages/huggingface_hub/utils/_token.py:89: 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),
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% 27.0/27.0 [00:00<00:00, 916B/s]
vocab.json: 100% 899k/899k [00:00<00:00, 3.61MB/s]
merges.txt: 100% 456k/456k [00:00<00:00, 7.91MB/s]
special_tokens_map.json: 100% 772/772 [00:00<00:00, 10.7kB/s]
config.json: 100% 1.09k/1.09k [00:00<00:00, 36.1kB/s]
/usr/local/lib/python3.10/dist-packages/transformers/tokenization_utils_base.py:1601: FutureWarning: `clean_up_tokenizat
  warnings.warn(
```

Data Spliting

```
import numpy as np
train, validate, test = np.split(tempdf.sample(frac=1, random_state=42), [int(.6*len(df)), int(.7*len(df))])
print(train.shape)
print(validate.shape)
print(test.shape)
validate = validate[:20]
print(validate.shape)
```

```
(7722, 4)
(1287, 4)
(3677, 4)
(20, 4)
/usr/local/lib/python3.10/dist-packages/numpy/core/fromnumeric.py:59: FutureWarning: 'DataFrame.swapaxes' is deprecated and will be removed in a
  return bound(*args, **kwds)
```

```
train_dataset = train_dataset.map(
    process_data_to_model_inputs,
    batched=True,
    batch_size=batch_size,
    remove_columns=["Article Title", "Subheading", "Paragraph", "length", "__index_level_0__"],
)

val_dataset = val_dataset.map(
    process_data_to_model_inputs,
    batched=True,
    batch_size=batch_size,
    remove_columns=["Article Title", "Subheading", "Paragraph", "length", "__index_level_0__"],
)

train_dataset.set_format(
    type="torch",
    columns=["input_ids", "attention_mask", "global_attention_mask", "labels"],
)

val_dataset.set_format(
    type="torch",
    columns=["input_ids", "attention_mask", "global_attention_mask", "labels"],
)
```

```
Map: 100% 7722/7722 [00:18<00:00, 606.21 examples/s]
Map: 100% 20/20 [00:00<00:00, 211.60 examples/s]
```

```
***** Running training *****
  Num examples = 7,722
  Num Epochs = 10
  Instantaneous batch size per device = 16
  Total train batch size (w. parallel, distributed & accumulation) = 64
  Gradient Accumulation steps = 4
  Total optimization steps = 1,200
  Number of trainable parameters = 161,844,480
/usr/local/lib/python3.10/dist-packages/torch/utils/checkpoint.py:295: FutureWarnir
  with torch.enable_grad(), device_autocast_ctx, torch.cpu.amp.autocast(**ctx.cpu_a
```

[81/1200 43:57 < 10:22:45, 0.03 it/s, Epoch 0.66/10]

Step	Training Loss	Validation Loss	Rouge2 Precision	Rouge2 Recall	Rouge2 Fmeasure
10	2.152900	1.785071	0.191700	0.181700	0.182400
20	2.129500	1.727933	0.191200	0.215300	0.199500
30	2.170000	1.719635	0.247100	0.263100	0.250000
40	2.153800	1.725903	0.212100	0.253000	0.217400
50	2.152200	1.740086	0.221900	0.249500	0.220900
60	2.043200	1.752729	0.286600	0.269000	0.270400
70	2.043300	1.717124	0.225600	0.226500	0.218900
80	2.151100	1.700823	0.252700	0.235800	0.237200

0s

```python
import pandas as pd
from datasets import Dataset, load_metric, load_dataset
from transformers import LEDTokenizer, LEDForConditionalGeneration
import torch

sample_paragraph = "Virat is an inspiration to many people around the world"
data = [sample_paragraph]
df = pd.DataFrame(data, columns=['Paragraph'])
df["Paragraph"][0]
df_test = Dataset.from_pandas(df)
df_test
```

```
Dataset({
    features: ['Paragraph'],
    num_rows: 1
})
```

0s

```python
result["generated_heading"]
```

```
['Virat']
```