text-summarization-and-generation

April 22, 2024

1 Installing Necessary Libraries

[1]:

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

```
Mounted at /content/drive
```

[2]:

```
!pip install transformers==4.12.2
!pip install bert-abstractive-summarizer
!pip install bert-extractive-summarizer
!pip install spacy==3.2.1
!python -m spacy download en_core_web_sm
```

```
Collecting transformers==4.12.2
  Downloading transformers-4.12.2-py3-none-any.whl (3.1 MB)
                           3.1/3.1 MB
11.6 MB/s eta 0:00:00
Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-
packages (from transformers==4.12.2) (3.13.4)
Requirement already satisfied: huggingface-hub>=0.0.17 in
/usr/local/lib/python3.10/dist-packages (from transformers==4.12.2) (0.20.3)
Requirement already satisfied: numpy>=1.17 in /usr/local/lib/python3.10/dist-
packages (from transformers==4.12.2) (1.25.2)
Requirement already satisfied: packaging>=20.0 in
/usr/local/lib/python3.10/dist-packages (from transformers==4.12.2) (24.0)
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.10/dist-
packages (from transformers==4.12.2) (6.0.1)
Requirement already satisfied: regex!=2019.12.17 in
/usr/local/lib/python3.10/dist-packages (from transformers==4.12.2) (2023.12.25)
Requirement already satisfied: requests in /usr/local/lib/python3.10/dist-
packages (from transformers==4.12.2) (2.31.0)
Collecting sacremoses (from transformers==4.12.2)
  Downloading sacremoses-0.1.1-py3-none-any.whl (897 kB)
                           897.5/897.5

kB 69.2 MB/s eta 0:00:00
Collecting tokenizers<0.11,>=0.10.1 (from transformers==4.12.2)
  Downloading tokenizers-0.10.3.tar.gz (212 kB)
```

```
                              212.7/212.7

kB 28.8 MB/s eta 0:00:00
  Installing build dependencies … done
  Getting requirements to build wheel … done
  Preparing metadata (pyproject.toml) … done
Requirement already satisfied: tqdm>=4.27 in /usr/local/lib/python3.10/dist-
packages (from transformers==4.12.2) (4.66.2)
Requirement already satisfied: fsspec>=2023.5.0 in
/usr/local/lib/python3.10/dist-packages (from huggingface-
hub>=0.0.17->transformers==4.12.2) (2023.6.0)
Requirement already satisfied: typing-extensions>=3.7.4.3 in
/usr/local/lib/python3.10/dist-packages (from huggingface-
hub>=0.0.17->transformers==4.12.2) (4.11.0)
Requirement already satisfied: charset-normalizer<4,>=2 in
/usr/local/lib/python3.10/dist-packages (from requests->transformers==4.12.2)
(3.3.2)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-
packages (from requests->transformers==4.12.2) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/usr/local/lib/python3.10/dist-packages (from requests->transformers==4.12.2)
(2.0.7)
Requirement already satisfied: certifi>=2017.4.17 in
/usr/local/lib/python3.10/dist-packages (from requests->transformers==4.12.2)
(2024.2.2)
Requirement already satisfied: click in /usr/local/lib/python3.10/dist-packages
(from sacremoses->transformers==4.12.2) (8.1.7)
Requirement already satisfied: joblib in /usr/local/lib/python3.10/dist-packages
(from sacremoses->transformers==4.12.2) (1.4.0)
Building wheels for collected packages: tokenizers
  error: subprocess-exited-with-error

  × Building wheel for tokenizers

(pyproject.toml) did not run successfully.
   exit code: 1
  ╰─> See above for output.

  note: This error originates from a subprocess, and is likely not a
problem with pip.
  Building wheel for tokenizers (pyproject.toml) … error
  ERROR: Failed building wheel for tokenizers

Failed to build tokenizers
```

```
ERROR: Could not build wheels for tokenizers, which is required to install
pyproject.toml-based projects
ERROR: Could not find a version that satisfies the requirement bert-
abstractive-summarizer (from versions: none)
ERROR: No matching distribution found for bert-abstractive-
summarizer
```

```
Collecting bert-extractive-summarizer
  Downloading bert_extractive_summarizer-0.10.1-py3-none-any.whl (25 kB)
Requirement already satisfied: transformers in /usr/local/lib/python3.10/dist-
packages (from bert-extractive-summarizer) (4.38.2)
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.10/dist-
packages (from bert-extractive-summarizer) (1.2.2)
Requirement already satisfied: spacy in /usr/local/lib/python3.10/dist-packages
(from bert-extractive-summarizer) (3.7.4)
Requirement already satisfied: numpy>=1.17.3 in /usr/local/lib/python3.10/dist-
packages (from scikit-learn->bert-extractive-summarizer) (1.25.2)
Requirement already satisfied: scipy>=1.3.2 in /usr/local/lib/python3.10/dist-
packages (from scikit-learn->bert-extractive-summarizer) (1.11.4)
Requirement already satisfied: joblib>=1.1.1 in /usr/local/lib/python3.10/dist-
packages (from scikit-learn->bert-extractive-summarizer) (1.4.0)
Requirement already satisfied: threadpoolctl>=2.0.0 in
/usr/local/lib/python3.10/dist-packages (from scikit-learn->bert-extractive-
summarizer) (3.4.0)
Requirement already satisfied: spacy-legacy<3.1.0,>=3.0.11 in
/usr/local/lib/python3.10/dist-packages (from spacy->bert-extractive-summarizer)
(3.0.12)
Requirement already satisfied: spacy-loggers<2.0.0,>=1.0.0 in
/usr/local/lib/python3.10/dist-packages (from spacy->bert-extractive-summarizer)
(1.0.5)
Requirement already satisfied: murmurhash<1.1.0,>=0.28.0 in
/usr/local/lib/python3.10/dist-packages (from spacy->bert-extractive-summarizer)
(1.0.10)
Requirement already satisfied: cymem<2.1.0,>=2.0.2 in
/usr/local/lib/python3.10/dist-packages (from spacy->bert-extractive-summarizer)
(2.0.8)
Requirement already satisfied: preshed<3.1.0,>=3.0.2 in
/usr/local/lib/python3.10/dist-packages (from spacy->bert-extractive-summarizer)
(3.0.9)
Requirement already satisfied: thinc<8.3.0,>=8.2.2 in
/usr/local/lib/python3.10/dist-packages (from spacy->bert-extractive-summarizer)
(8.2.3)
Requirement already satisfied: wasabi<1.2.0,>=0.9.1 in
/usr/local/lib/python3.10/dist-packages (from spacy->bert-extractive-summarizer)
(1.1.2)
```

```
Requirement already satisfied: srsly<3.0.0,>=2.4.3 in
/usr/local/lib/python3.10/dist-packages (from spacy->bert-extractive-summarizer)
(2.4.8)
Requirement already satisfied: catalogue<2.1.0,>=2.0.6 in
/usr/local/lib/python3.10/dist-packages (from spacy->bert-extractive-summarizer)
(2.0.10)
Requirement already satisfied: weasel<0.4.0,>=0.1.0 in
/usr/local/lib/python3.10/dist-packages (from spacy->bert-extractive-summarizer)
(0.3.4)
Requirement already satisfied: typer<0.10.0,>=0.3.0 in
/usr/local/lib/python3.10/dist-packages (from spacy->bert-extractive-summarizer)
(0.9.4)
Requirement already satisfied: smart-open<7.0.0,>=5.2.1 in
/usr/local/lib/python3.10/dist-packages (from spacy->bert-extractive-summarizer)
(6.4.0)
Requirement already satisfied: tqdm<5.0.0,>=4.38.0 in
/usr/local/lib/python3.10/dist-packages (from spacy->bert-extractive-summarizer)
(4.66.2)
Requirement already satisfied: requests<3.0.0,>=2.13.0 in
/usr/local/lib/python3.10/dist-packages (from spacy->bert-extractive-summarizer)
(2.31.0)
Requirement already satisfied: pydantic!=1.8,!=1.8.1,<3.0.0,>=1.7.4 in
/usr/local/lib/python3.10/dist-packages (from spacy->bert-extractive-summarizer)
(2.7.0)
Requirement already satisfied: jinja2 in /usr/local/lib/python3.10/dist-packages
(from spacy->bert-extractive-summarizer) (3.1.3)
Requirement already satisfied: setuptools in /usr/local/lib/python3.10/dist-
packages (from spacy->bert-extractive-summarizer) (67.7.2)
Requirement already satisfied: packaging>=20.0 in
/usr/local/lib/python3.10/dist-packages (from spacy->bert-extractive-summarizer)
(24.0)
Requirement already satisfied: langcodes<4.0.0,>=3.2.0 in
/usr/local/lib/python3.10/dist-packages (from spacy->bert-extractive-summarizer)
(3.3.0)
Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-
packages (from transformers->bert-extractive-summarizer) (3.13.4)
Requirement already satisfied: huggingface-hub<1.0,>=0.19.3 in
/usr/local/lib/python3.10/dist-packages (from transformers->bert-extractive-
summarizer) (0.20.3)
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.10/dist-
packages (from transformers->bert-extractive-summarizer) (6.0.1)
Requirement already satisfied: regex!=2019.12.17 in
/usr/local/lib/python3.10/dist-packages (from transformers->bert-extractive-
summarizer) (2023.12.25)
Requirement already satisfied: tokenizers<0.19,>=0.14 in
/usr/local/lib/python3.10/dist-packages (from transformers->bert-extractive-
summarizer) (0.15.2)
Requirement already satisfied: safetensors>=0.4.1 in
```

```
/usr/local/lib/python3.10/dist-packages (from transformers->bert-extractive-
summarizer) (0.4.3)
Requirement already satisfied: fsspec>=2023.5.0 in
/usr/local/lib/python3.10/dist-packages (from huggingface-
hub<1.0,>=0.19.3->transformers->bert-extractive-summarizer) (2023.6.0)
Requirement already satisfied: typing-extensions>=3.7.4.3 in
/usr/local/lib/python3.10/dist-packages (from huggingface-
hub<1.0,>=0.19.3->transformers->bert-extractive-summarizer) (4.11.0)
Requirement already satisfied: annotated-types>=0.4.0 in
/usr/local/lib/python3.10/dist-packages (from
pydantic!=1.8,!=1.8.1,<3.0.0,>=1.7.4->spacy->bert-extractive-summarizer) (0.6.0)
Requirement already satisfied: pydantic-core==2.18.1 in
/usr/local/lib/python3.10/dist-packages (from
pydantic!=1.8,!=1.8.1,<3.0.0,>=1.7.4->spacy->bert-extractive-summarizer)
(2.18.1)
Requirement already satisfied: charset-normalizer<4,>=2 in
/usr/local/lib/python3.10/dist-packages (from
requests<3.0.0,>=2.13.0->spacy->bert-extractive-summarizer) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-
packages (from requests<3.0.0,>=2.13.0->spacy->bert-extractive-summarizer) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/usr/local/lib/python3.10/dist-packages (from
requests<3.0.0,>=2.13.0->spacy->bert-extractive-summarizer) (2.0.7)
Requirement already satisfied: certifi>=2017.4.17 in
/usr/local/lib/python3.10/dist-packages (from
requests<3.0.0,>=2.13.0->spacy->bert-extractive-summarizer) (2024.2.2)
Requirement already satisfied: blis<0.8.0,>=0.7.8 in
/usr/local/lib/python3.10/dist-packages (from thinc<8.3.0,>=8.2.2->spacy->bert-
extractive-summarizer) (0.7.11)
Requirement already satisfied: confection<1.0.0,>=0.0.1 in
/usr/local/lib/python3.10/dist-packages (from thinc<8.3.0,>=8.2.2->spacy->bert-
extractive-summarizer) (0.1.4)
Requirement already satisfied: click<9.0.0,>=7.1.1 in
/usr/local/lib/python3.10/dist-packages (from typer<0.10.0,>=0.3.0->spacy->bert-
extractive-summarizer) (8.1.7)
Requirement already satisfied: cloudpathlib<0.17.0,>=0.7.0 in
/usr/local/lib/python3.10/dist-packages (from weasel<0.4.0,>=0.1.0->spacy->bert-
extractive-summarizer) (0.16.0)
Requirement already satisfied: MarkupSafe>=2.0 in
/usr/local/lib/python3.10/dist-packages (from jinja2->spacy->bert-extractive-
summarizer) (2.1.5)
Installing collected packages: bert-extractive-summarizer
Successfully installed bert-extractive-summarizer-0.10.1
Collecting spacy==3.2.1
  Downloading
spacy-3.2.1-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (6.1 MB)
                           6.1/6.1 MB
22.3 MB/s eta 0:00:00
```

```
Requirement already satisfied: spacy-legacy<3.1.0,>=3.0.8 in
/usr/local/lib/python3.10/dist-packages (from spacy==3.2.1) (3.0.12)
Requirement already satisfied: spacy-loggers<2.0.0,>=1.0.0 in
/usr/local/lib/python3.10/dist-packages (from spacy==3.2.1) (1.0.5)
Requirement already satisfied: murmurhash<1.1.0,>=0.28.0 in
/usr/local/lib/python3.10/dist-packages (from spacy==3.2.1) (1.0.10)
Requirement already satisfied: cymem<2.1.0,>=2.0.2 in
/usr/local/lib/python3.10/dist-packages (from spacy==3.2.1) (2.0.8)
Requirement already satisfied: preshed<3.1.0,>=3.0.2 in
/usr/local/lib/python3.10/dist-packages (from spacy==3.2.1) (3.0.9)
Collecting thinc<8.1.0,>=8.0.12 (from spacy==3.2.1)
  Downloading
thinc-8.0.17-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (659 kB)
                           659.5/659.5

kB 62.6 MB/s eta 0:00:00
Requirement already satisfied: blis<0.8.0,>=0.4.0 in
/usr/local/lib/python3.10/dist-packages (from spacy==3.2.1) (0.7.11)
Collecting wasabi<1.1.0,>=0.8.1 (from spacy==3.2.1)
  Downloading wasabi-0.10.1-py3-none-any.whl (26 kB)
Requirement already satisfied: srsly<3.0.0,>=2.4.1 in
/usr/local/lib/python3.10/dist-packages (from spacy==3.2.1) (2.4.8)
Requirement already satisfied: catalogue<2.1.0,>=2.0.6 in
/usr/local/lib/python3.10/dist-packages (from spacy==3.2.1) (2.0.10)
Collecting typer<0.5.0,>=0.3.0 (from spacy==3.2.1)
  Downloading typer-0.4.2-py3-none-any.whl (27 kB)
Collecting pathy>=0.3.5 (from spacy==3.2.1)
  Downloading pathy-0.11.0-py3-none-any.whl (47 kB)
                           47.3/47.3 kB
7.8 MB/s eta 0:00:00
Requirement already satisfied: tqdm<5.0.0,>=4.38.0 in
/usr/local/lib/python3.10/dist-packages (from spacy==3.2.1) (4.66.2)
Requirement already satisfied: numpy>=1.15.0 in /usr/local/lib/python3.10/dist-
packages (from spacy==3.2.1) (1.25.2)
Requirement already satisfied: requests<3.0.0,>=2.13.0 in
/usr/local/lib/python3.10/dist-packages (from spacy==3.2.1) (2.31.0)
Collecting pydantic!=1.8,!=1.8.1,<1.9.0,>=1.7.4 (from spacy==3.2.1)
  Downloading pydantic-1.8.2-py3-none-any.whl (126 kB)
                           126.0/126.0

kB 16.9 MB/s eta 0:00:00
Requirement already satisfied: jinja2 in /usr/local/lib/python3.10/dist-
packages (from spacy==3.2.1) (3.1.3)
Requirement already satisfied: setuptools in /usr/local/lib/python3.10/dist-
packages (from spacy==3.2.1) (67.7.2)
Requirement already satisfied: packaging>=20.0 in
/usr/local/lib/python3.10/dist-packages (from spacy==3.2.1) (24.0)
Requirement already satisfied: langcodes<4.0.0,>=3.2.0 in
```

```
/usr/local/lib/python3.10/dist-packages (from spacy==3.2.1) (3.3.0)
Requirement already satisfied: smart-open<7.0.0,>=5.2.1 in
/usr/local/lib/python3.10/dist-packages (from pathy>=0.3.5->spacy==3.2.1)
(6.4.0)
Collecting pathlib-abc==0.1.1 (from pathy>=0.3.5->spacy==3.2.1)
  Downloading pathlib_abc-0.1.1-py3-none-any.whl (23 kB)
Requirement already satisfied: typing-extensions>=3.7.4.3 in
/usr/local/lib/python3.10/dist-packages (from
pydantic!=1.8,!=1.8.1,<1.9.0,>=1.7.4->spacy==3.2.1) (4.11.0)
Requirement already satisfied: charset-normalizer<4,>=2 in
/usr/local/lib/python3.10/dist-packages (from
requests<3.0.0,>=2.13.0->spacy==3.2.1) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-
packages (from requests<3.0.0,>=2.13.0->spacy==3.2.1) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/usr/local/lib/python3.10/dist-packages (from
requests<3.0.0,>=2.13.0->spacy==3.2.1) (2.0.7)
Requirement already satisfied: certifi>=2017.4.17 in
/usr/local/lib/python3.10/dist-packages (from
requests<3.0.0,>=2.13.0->spacy==3.2.1) (2024.2.2)
Requirement already satisfied: click<9.0.0,>=7.1.1 in
/usr/local/lib/python3.10/dist-packages (from typer<0.5.0,>=0.3.0->spacy==3.2.1)
(8.1.7)
Requirement already satisfied: MarkupSafe>=2.0 in
/usr/local/lib/python3.10/dist-packages (from jinja2->spacy==3.2.1) (2.1.5)
Installing collected packages: wasabi, typer, pydantic, pathlib-abc, thinc,
pathy, spacy
  Attempting uninstall: wasabi
    Found existing installation: wasabi 1.1.2
    Uninstalling wasabi-1.1.2:
      Successfully uninstalled wasabi-1.1.2
  Attempting uninstall: typer
    Found existing installation: typer 0.9.4
    Uninstalling typer-0.9.4:
      Successfully uninstalled typer-0.9.4
  Attempting uninstall: pydantic
    Found existing installation: pydantic 2.7.0
    Uninstalling pydantic-2.7.0:
      Successfully uninstalled pydantic-2.7.0
  Attempting uninstall: thinc
    Found existing installation: thinc 8.2.3
    Uninstalling thinc-8.2.3:
      Successfully uninstalled thinc-8.2.3
  Attempting uninstall: spacy
    Found existing installation: spacy 3.7.4
    Uninstalling spacy-3.7.4:
      Successfully uninstalled spacy-3.7.4
```

```
ERROR: pip's dependency resolver does not currently take into account all
the packages that are installed. This behaviour is the source of the following
dependency conflicts.
en-core-web-sm 3.7.1 requires spacy<3.8.0,>=3.7.2, but you have spacy 3.2.1
which is incompatible.
inflect 7.0.0 requires pydantic>=1.9.1, but you have pydantic 1.8.2 which is
incompatible.
Successfully installed pathlib-abc-0.1.1 pathy-0.11.0 pydantic-1.8.2
spacy-3.2.1 thinc-8.0.17 typer-0.4.2 wasabi-0.10.1
2024-04-22 08:55:22.642038: E
external/local_xla/xla/stream_executor/cuda/cuda_dnn.cc:9261] Unable to register
cuDNN factory: Attempting to register factory for plugin cuDNN when one has
already been registered
2024-04-22 08:55:22.642094: E
external/local_xla/xla/stream_executor/cuda/cuda_fft.cc:607] Unable to register
cuFFT factory: Attempting to register factory for plugin cuFFT when one has
already been registered
2024-04-22 08:55:22.643406: E
external/local_xla/xla/stream_executor/cuda/cuda_blas.cc:1515] Unable to
register cuBLAS factory: Attempting to register factory for plugin cuBLAS when
one has already been registered
2024-04-22 08:55:22.650677: I tensorflow/core/platform/cpu_feature_guard.cc:182]
This TensorFlow binary is optimized to use available CPU instructions in
performance-critical operations.
To enable the following instructions: AVX2 FMA, in other operations, rebuild
TensorFlow with the appropriate compiler flags.
2024-04-22 08:55:23.620322: W
tensorflow/compiler/tf2tensorrt/utils/py_utils.cc:38] TF-TRT Warning: Could not
find TensorRT
DEPRECATION: https://github.com/explosion/spacy-
models/releases/download/en_core_web_sm-3.2.0/en_core_web_sm-3.2.0-py3-none-
any.whl#egg=en_core_web_sm==3.2.0 contains an egg fragment with a non-PEP 508
name pip 25.0 will enforce this behaviour change. A possible replacement is to
use the req @ url syntax, and remove the egg fragment. Discussion can be found
at https://github.com/pypa/pip/issues/11617
Collecting en-core-web-sm==3.2.0
  Downloading https://github.com/explosion/spacy-
models/releases/download/en_core_web_sm-3.2.0/en_core_web_sm-3.2.0-py3-none-
any.whl (13.9 MB)
                           13.9/13.9 MB
35.6 MB/s eta 0:00:00
```

```
Requirement already satisfied: spacy<3.3.0,>=3.2.0 in
/usr/local/lib/python3.10/dist-packages (from en-core-web-sm==3.2.0) (3.2.1)
Requirement already satisfied: spacy-legacy<3.1.0,>=3.0.8 in
/usr/local/lib/python3.10/dist-packages (from spacy<3.3.0,>=3.2.0->en-core-web-
sm==3.2.0) (3.0.12)
Requirement already satisfied: spacy-loggers<2.0.0,>=1.0.0 in
/usr/local/lib/python3.10/dist-packages (from spacy<3.3.0,>=3.2.0->en-core-web-
sm==3.2.0) (1.0.5)
Requirement already satisfied: murmurhash<1.1.0,>=0.28.0 in
/usr/local/lib/python3.10/dist-packages (from spacy<3.3.0,>=3.2.0->en-core-web-
sm==3.2.0) (1.0.10)
Requirement already satisfied: cymem<2.1.0,>=2.0.2 in
/usr/local/lib/python3.10/dist-packages (from spacy<3.3.0,>=3.2.0->en-core-web-
sm==3.2.0) (2.0.8)
Requirement already satisfied: preshed<3.1.0,>=3.0.2 in
/usr/local/lib/python3.10/dist-packages (from spacy<3.3.0,>=3.2.0->en-core-web-
sm==3.2.0) (3.0.9)
Requirement already satisfied: thinc<8.1.0,>=8.0.12 in
/usr/local/lib/python3.10/dist-packages (from spacy<3.3.0,>=3.2.0->en-core-web-
sm==3.2.0) (8.0.17)
Requirement already satisfied: blis<0.8.0,>=0.4.0 in
/usr/local/lib/python3.10/dist-packages (from spacy<3.3.0,>=3.2.0->en-core-web-
sm==3.2.0) (0.7.11)
Requirement already satisfied: wasabi<1.1.0,>=0.8.1 in
/usr/local/lib/python3.10/dist-packages (from spacy<3.3.0,>=3.2.0->en-core-web-
sm==3.2.0) (0.10.1)
Requirement already satisfied: srsly<3.0.0,>=2.4.1 in
/usr/local/lib/python3.10/dist-packages (from spacy<3.3.0,>=3.2.0->en-core-web-
sm==3.2.0) (2.4.8)
Requirement already satisfied: catalogue<2.1.0,>=2.0.6 in
/usr/local/lib/python3.10/dist-packages (from spacy<3.3.0,>=3.2.0->en-core-web-
sm==3.2.0) (2.0.10)
Requirement already satisfied: typer<0.5.0,>=0.3.0 in
/usr/local/lib/python3.10/dist-packages (from spacy<3.3.0,>=3.2.0->en-core-web-
sm==3.2.0) (0.4.2)
Requirement already satisfied: pathy>=0.3.5 in /usr/local/lib/python3.10/dist-
packages (from spacy<3.3.0,>=3.2.0->en-core-web-sm==3.2.0) (0.11.0)
Requirement already satisfied: tqdm<5.0.0,>=4.38.0 in
/usr/local/lib/python3.10/dist-packages (from spacy<3.3.0,>=3.2.0->en-core-web-
sm==3.2.0) (4.66.2)
Requirement already satisfied: numpy>=1.15.0 in /usr/local/lib/python3.10/dist-
packages (from spacy<3.3.0,>=3.2.0->en-core-web-sm==3.2.0) (1.25.2)
Requirement already satisfied: requests<3.0.0,>=2.13.0 in
/usr/local/lib/python3.10/dist-packages (from spacy<3.3.0,>=3.2.0->en-core-web-
sm==3.2.0) (2.31.0)
Requirement already satisfied: pydantic!=1.8,!=1.8.1,<1.9.0,>=1.7.4 in
/usr/local/lib/python3.10/dist-packages (from spacy<3.3.0,>=3.2.0->en-core-web-
sm==3.2.0) (1.8.2)
```

```
Requirement already satisfied: jinja2 in /usr/local/lib/python3.10/dist-packages
(from spacy<3.3.0,>=3.2.0->en-core-web-sm==3.2.0) (3.1.3)
Requirement already satisfied: setuptools in /usr/local/lib/python3.10/dist-
packages (from spacy<3.3.0,>=3.2.0->en-core-web-sm==3.2.0) (67.7.2)
Requirement already satisfied: packaging>=20.0 in
/usr/local/lib/python3.10/dist-packages (from spacy<3.3.0,>=3.2.0->en-core-web-
sm==3.2.0) (24.0)
Requirement already satisfied: langcodes<4.0.0,>=3.2.0 in
/usr/local/lib/python3.10/dist-packages (from spacy<3.3.0,>=3.2.0->en-core-web-
sm==3.2.0) (3.3.0)
Requirement already satisfied: smart-open<7.0.0,>=5.2.1 in
/usr/local/lib/python3.10/dist-packages (from
pathy>=0.3.5->spacy<3.3.0,>=3.2.0->en-core-web-sm==3.2.0) (6.4.0)
Requirement already satisfied: pathlib-abc==0.1.1 in
/usr/local/lib/python3.10/dist-packages (from
pathy>=0.3.5->spacy<3.3.0,>=3.2.0->en-core-web-sm==3.2.0) (0.1.1)
Requirement already satisfied: typing-extensions>=3.7.4.3 in
/usr/local/lib/python3.10/dist-packages (from
pydantic!=1.8,!=1.8.1,<1.9.0,>=1.7.4->spacy<3.3.0,>=3.2.0->en-core-web-
sm==3.2.0) (4.11.0)
Requirement already satisfied: charset-normalizer<4,>=2 in
/usr/local/lib/python3.10/dist-packages (from
requests<3.0.0,>=2.13.0->spacy<3.3.0,>=3.2.0->en-core-web-sm==3.2.0) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-
packages (from requests<3.0.0,>=2.13.0->spacy<3.3.0,>=3.2.0->en-core-web-
sm==3.2.0) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/usr/local/lib/python3.10/dist-packages (from
requests<3.0.0,>=2.13.0->spacy<3.3.0,>=3.2.0->en-core-web-sm==3.2.0) (2.0.7)
Requirement already satisfied: certifi>=2017.4.17 in
/usr/local/lib/python3.10/dist-packages (from
requests<3.0.0,>=2.13.0->spacy<3.3.0,>=3.2.0->en-core-web-sm==3.2.0) (2024.2.2)
Requirement already satisfied: click<9.0.0,>=7.1.1 in
/usr/local/lib/python3.10/dist-packages (from
typer<0.5.0,>=0.3.0->spacy<3.3.0,>=3.2.0->en-core-web-sm==3.2.0) (8.1.7)
Requirement already satisfied: MarkupSafe>=2.0 in
/usr/local/lib/python3.10/dist-packages (from jinja2->spacy<3.3.0,>=3.2.0->en-
core-web-sm==3.2.0) (2.1.5)
Installing collected packages: en-core-web-sm
  Attempting uninstall: en-core-web-sm
    Found existing installation: en-core-web-sm 3.7.1
    Uninstalling en-core-web-sm-3.7.1:
      Successfully uninstalled en-core-web-sm-3.7.1
Successfully installed en-core-web-sm-3.2.0
 Download and installation successful
You can now load the package via spacy.load('en_core_web_sm')
```

2 Import Summarizer

[3]:
```
import pandas as pd
import numpy as np
```

[4]:
```
!pip install python-docx
```

```
Collecting python-docx
  Downloading python_docx-1.1.0-py3-none-any.whl (239 kB)
                           239.6/239.6

kB 2.4 MB/s eta 0:00:00
Requirement already satisfied: lxml>=3.1.0 in
/usr/local/lib/python3.10/dist-packages (from python-docx) (4.9.4)
Requirement already satisfied: typing-extensions in
/usr/local/lib/python3.10/dist-packages (from python-docx) (4.11.0)
Installing collected packages: python-docx
Successfully installed python-docx-1.1.0
```

[5]:
```
import warnings

warnings.simplefilter(action='ignore', category = FutureWarning)
```

[6]:
```
from summarizer import Summarizer, TransformerSummarizer
```

```
/usr/local/lib/python3.10/dist-packages/torch/__init__.py:696: UserWarning:
torch.set_default_tensor_type() is deprecated as of PyTorch 2.1, please use
torch.set_default_dtype() and torch.set_default_device() as alternatives.
(Triggered internally at ../torch/csrc/tensor/python_tensor.cpp:451.)
  _C._set_default_tensor_type(t)
```

3 Define the article body

[7]:
```
from docx import Document

def read_docx(file_path):
    doc = Document(file_path)
    text = ""
    for paragraph in doc.paragraphs:
        text += paragraph.text + "\n"
    return text

# Example usage
file_path = "/content/drive/MyDrive/Text_Summarization_and_Generation/Article.
 ↪docx"
art = read_docx(file_path)
print(art)
```

Title: Breakthrough in Renewable Energy: Novel Solar Technology Promises Game-Changing Efficiency

Introduction:
In a world increasingly grappling with climate change, the quest for sustainable and efficient energy sources has never been more critical. The latest breakthrough in renewable energy comes in the form of revolutionary solar technology that promises to transform the landscape of clean energy production. Developed by a team of researchers at [Institution/Company], this innovation boasts unprecedented efficiency levels that could potentially revolutionize the global energy sector.

Background:
Solar energy has long been hailed as a promising solution to the world's energy needs. Traditional solar panels have made significant strides in recent years, but their efficiency and scalability still pose challenges. However, the emergence of next-generation solar technologies offers renewed hope for a future powered by clean, renewable energy.

The Breakthrough:
The breakthrough technology, dubbed "SolarMax," represents a significant leap forward in solar energy efficiency. Unlike conventional solar panels that rely on photovoltaic cells to convert sunlight into electricity, SolarMax utilizes a novel approach inspired by [describe the inspiration/source of innovation, e.g., biological processes or quantum mechanics].

At the heart of SolarMax lies a sophisticated nanomaterial that exhibits exceptional light-absorbing properties. This material, developed through years of research and experimentation, allows SolarMax to capture a broader spectrum of sunlight, including wavelengths that traditional solar panels cannot harness effectively. As a result, SolarMax achieves an unprecedented efficiency rating of over 40%, far surpassing the capabilities of current solar technologies.

Key Features and Advantages:
1. **Enhanced Efficiency:** SolarMax's groundbreaking design enables it to convert a higher percentage of sunlight into electricity, maximizing energy output and optimizing space utilization.
2. **Versatility:** SolarMax is adaptable to various environments and applications, from residential rooftops to large-scale solar farms, making it a versatile solution for diverse energy needs.
3. **Durability:** The nanomaterial used in SolarMax is highly durable and resilient, ensuring long-term performance and minimal maintenance requirements.
4. **Scalability:** The modular design of SolarMax facilitates easy scalability, allowing for seamless integration into existing solar infrastructure or the development of new solar projects of any size.

Potential Impact:
The implications of SolarMax extend far beyond its impressive technical

specifications. With its unparalleled efficiency and versatility, this breakthrough technology has the potential to revolutionize the global energy landscape in several ways:

1. **Accelerated Transition to Renewable Energy:** By significantly increasing the efficiency of solar energy production, SolarMax could expedite the transition away from fossil fuels, reducing greenhouse gas emissions and mitigating the effects of climate change.
2. **Energy Independence:** SolarMax's widespread adoption could enhance energy independence for countries, reducing reliance on imported fossil fuels and strengthening energy security.
3. **Economic Opportunities:** The development and deployment of SolarMax could create new economic opportunities, driving job growth in the renewable energy sector and stimulating innovation and investment.
4. **Access to Clean Energy:** SolarMax's scalability and affordability could improve access to clean energy in underserved communities, empowering individuals and regions with sustainable power sources.

Challenges and Future Directions:
While SolarMax represents a remarkable achievement in renewable energy technology, several challenges and considerations remain:

1. **Cost:** Despite its impressive efficiency, SolarMax's initial manufacturing and installation costs may present barriers to widespread adoption, particularly in developing countries or regions with limited resources.
2. **Technological Advancements:** Continued research and development are essential to further optimize SolarMax's performance, reduce production costs, and address any potential environmental or safety concerns associated with its use.
3. **Integration and Infrastructure:** The integration of SolarMax into existing energy infrastructure and regulatory frameworks will require careful planning and coordination to maximize its benefits and ensure a smooth transition.
4. **Global Cooperation:** Achieving the full potential of SolarMax and other renewable energy technologies will require international cooperation and collaboration to overcome political, economic, and logistical challenges.

Conclusion:
The development of SolarMax represents a significant milestone in the pursuit of sustainable energy solutions. With its unprecedented efficiency and versatility, this breakthrough technology holds the promise of a cleaner, more resilient energy future for generations to come. As researchers continue to push the boundaries of innovation and technology, SolarMax serves as a shining example of human ingenuity and determination in the face of one of the greatest challenges of our time: combating climate change and securing a sustainable planet for future generations.

[9]:

```
Context = art
```

[10]:

```
print(Context)
```

```
**Title: Breakthrough in Renewable Energy: Novel Solar Technology Promises Game-
Changing Efficiency**

**Introduction:**
In a world increasingly grappling with climate change, the quest for sustainable
and efficient energy sources has never been more critical. The latest
breakthrough in renewable energy comes in the form of revolutionary solar
technology that promises to transform the landscape of clean energy production.
Developed by a team of researchers at [Institution/Company], this innovation
boasts unprecedented efficiency levels that could potentially revolutionize the
global energy sector.

**Background:**
Solar energy has long been hailed as a promising solution to the world's energy
needs. Traditional solar panels have made significant strides in recent years,
but their efficiency and scalability still pose challenges. However, the
emergence of next-generation solar technologies offers renewed hope for a future
powered by clean, renewable energy.

**The Breakthrough:**
The breakthrough technology, dubbed "SolarMax," represents a significant leap
forward in solar energy efficiency. Unlike conventional solar panels that rely
on photovoltaic cells to convert sunlight into electricity, SolarMax utilizes a
novel approach inspired by [describe the inspiration/source of innovation, e.g.,
biological processes or quantum mechanics].

At the heart of SolarMax lies a sophisticated nanomaterial that exhibits
exceptional light-absorbing properties. This material, developed through years
of research and experimentation, allows SolarMax to capture a broader spectrum
of sunlight, including wavelengths that traditional solar panels cannot harness
effectively. As a result, SolarMax achieves an unprecedented efficiency rating
of over 40%, far surpassing the capabilities of current solar technologies.

**Key Features and Advantages:**
1. **Enhanced Efficiency:** SolarMax's groundbreaking design enables it to
convert a higher percentage of sunlight into electricity, maximizing energy
output and optimizing space utilization.
2. **Versatility:** SolarMax is adaptable to various environments and
applications, from residential rooftops to large-scale solar farms, making it a
versatile solution for diverse energy needs.
3. **Durability:** The nanomaterial used in SolarMax is highly durable and
resilient, ensuring long-term performance and minimal maintenance requirements.
4. **Scalability:** The modular design of SolarMax facilitates easy scalability,
```

allowing for seamless integration into existing solar infrastructure or the development of new solar projects of any size.

Potential Impact:
The implications of SolarMax extend far beyond its impressive technical specifications. With its unparalleled efficiency and versatility, this breakthrough technology has the potential to revolutionize the global energy landscape in several ways:

1. **Accelerated Transition to Renewable Energy:** By significantly increasing the efficiency of solar energy production, SolarMax could expedite the transition away from fossil fuels, reducing greenhouse gas emissions and mitigating the effects of climate change.
2. **Energy Independence:** SolarMax's widespread adoption could enhance energy independence for countries, reducing reliance on imported fossil fuels and strengthening energy security.
3. **Economic Opportunities:** The development and deployment of SolarMax could create new economic opportunities, driving job growth in the renewable energy sector and stimulating innovation and investment.
4. **Access to Clean Energy:** SolarMax's scalability and affordability could improve access to clean energy in underserved communities, empowering individuals and regions with sustainable power sources.

Challenges and Future Directions:
While SolarMax represents a remarkable achievement in renewable energy technology, several challenges and considerations remain:

1. **Cost:** Despite its impressive efficiency, SolarMax's initial manufacturing and installation costs may present barriers to widespread adoption, particularly in developing countries or regions with limited resources.
2. **Technological Advancements:** Continued research and development are essential to further optimize SolarMax's performance, reduce production costs, and address any potential environmental or safety concerns associated with its use.
3. **Integration and Infrastructure:** The integration of SolarMax into existing energy infrastructure and regulatory frameworks will require careful planning and coordination to maximize its benefits and ensure a smooth transition.
4. **Global Cooperation:** Achieving the full potential of SolarMax and other renewable energy technologies will require international cooperation and collaboration to overcome political, economic, and logistical challenges.

Conclusion:
The development of SolarMax represents a significant milestone in the pursuit of sustainable energy solutions. With its unprecedented efficiency and versatility, this breakthrough technology holds the promise of a cleaner, more resilient energy future for generations to come. As researchers continue to push the boundaries of innovation and technology, SolarMax serves as a shining example of human ingenuity and determination in the face of one of the greatest challenges

```
of our time: combating climate change and securing a sustainable planet for
future generations.
```

4 Summary of original Context

[11]:

```python
from heapq import nlargest
from collections import Counter
import re

def read_article(article):
    sentences = re.split(r'(?<!\w\.\w.)(?<![A-Z][a-z]\.)(?<=\.|\?)\s', article)
    return sentences

def sentence_similarity(sent1, sent2):
    words1 = sent1.split()
    words2 = sent2.split()
    common_words = set(words1) & set(words2)
    return len(common_words)

def build_similarity_matrix(sentences):
    similarity_matrix = [[0 for _ in range(len(sentences))] for _ in␣
 ↪range(len(sentences))]
    for i in range(len(sentences)):
        for j in range(len(sentences)):
            if i!= j:
                similarity_matrix[i][j] = sentence_similarity(sentences[i],␣
 ↪sentences[j])
    return similarity_matrix

def generate_summary(article, top_n=5):
    summarized_text = ""
    sentences = read_article(article)
    sentence_similarity_matrix = build_similarity_matrix(sentences)
    sentence_similarity_scores = Counter()

    for i in range(len(sentences)):
        sentence_similarity_scores[i] = sum(sentence_similarity_matrix[i])

    summarized_sentences = nlargest(top_n, sentence_similarity_scores,␣
 ↪key=sentence_similarity_scores.get)

    for sentence_index in summarized_sentences:
        summarized_text += sentences[sentence_index] + " "
    return summarized_text
```

[12]:

```
# Example usage
article = """
User
*Title: Breakthrough in Renewable Energy: Novel Solar Technology Promises␣
 ↪Game-Changing Efficiency**

**Introduction:**
In a world increasingly grappling with climate change, the quest for␣
 ↪sustainable and efficient energy sources has never been more critical. The␣
 ↪latest breakthrough in renewable energy comes in the form of revolutionary␣
 ↪solar technology that promises to transform the landscape of clean energy␣
 ↪production. Developed by a team of researchers at [Institution/Company],␣
 ↪this innovation boasts unprecedented efficiency levels that could␣
 ↪potentially revolutionize the global energy sector.

**Background:**
Solar energy has long been hailed as a promising solution to the world's energy␣
 ↪needs. Traditional solar panels have made significant strides in recent␣
 ↪years, but their efficiency and scalability still pose challenges. However,␣
 ↪the emergence of next-generation solar technologies offers renewed hope for␣
 ↪a future powered by clean, renewable energy.

**The Breakthrough:**
The breakthrough technology, dubbed "SolarMax," represents a significant leap␣
 ↪forward in solar energy efficiency. Unlike conventional solar panels that␣
 ↪rely on photovoltaic cells to convert sunlight into electricity, SolarMax␣
 ↪utilizes a novel approach inspired by [describe the inspiration/source of␣
 ↪innovation, e.g., biological processes or quantum mechanics].

At the heart of SolarMax lies a sophisticated nanomaterial that exhibits␣
 ↪exceptional light-absorbing properties. This material, developed through␣
 ↪years of research and experimentation, allows SolarMax to capture a broader␣
 ↪spectrum of sunlight, including wavelengths that traditional solar panels␣
 ↪cannot harness effectively. As a result, SolarMax achieves an unprecedented␣
 ↪efficiency rating of over 40%, far surpassing the capabilities of current␣
 ↪solar technologies.

**Key Features and Advantages:**
1. **Enhanced Efficiency:** SolarMax's groundbreaking design enables it to␣
 ↪convert a higher percentage of sunlight into electricity, maximizing energy␣
 ↪output and optimizing space utilization.
2. **Versatility:** SolarMax is adaptable to various environments and␣
 ↪applications, from residential rooftops to large-scale solar farms, making␣
 ↪it a versatile solution for diverse energy needs.
```

```
3. **Durability:** The nanomaterial used in SolarMax is highly durable and␣
 ↪resilient, ensuring long-term performance and minimal maintenance␣
 ↪requirements.
4. **Scalability:** The modular design of SolarMax facilitates easy␣
 ↪scalability, allowing for seamless integration into existing solar␣
 ↪infrastructure or the development of new solar projects of any size.

**Potential Impact:**
The implications of SolarMax extend far beyond its impressive technical␣
 ↪specifications. With its unparalleled efficiency and versatility, this␣
 ↪breakthrough technology has the potential to revolutionize the global energy␣
 ↪landscape in several ways:

1. **Accelerated Transition to Renewable Energy:** By significantly increasing␣
 ↪the efficiency of solar energy production, SolarMax could expedite the␣
 ↪transition away from fossil fuels, reducing greenhouse gas emissions and␣
 ↪mitigating the effects of climate change.
2. **Energy Independence:** SolarMax's widespread adoption could enhance energy␣
 ↪independence for countries, reducing reliance on imported fossil fuels and␣
 ↪strengthening energy security.
3. **Economic Opportunities:** The development and deployment of SolarMax could␣
 ↪create new economic opportunities, driving job growth in the renewable␣
 ↪energy sector and stimulating innovation and investment.
4. **Access to Clean Energy:** SolarMax's scalability and affordability could␣
 ↪improve access to clean energy in underserved communities, empowering␣
 ↪individuals and regions with sustainable power sources.

**Challenges and Future Directions:**
While SolarMax represents a remarkable achievement in renewable energy␣
 ↪technology, several challenges and considerations remain:

1. **Cost:** Despite its impressive efficiency, SolarMax's initial␣
 ↪manufacturing and installation costs may present barriers to widespread␣
 ↪adoption, particularly in developing countries or regions with limited␣
 ↪resources.
2. **Technological Advancements:** Continued research and development are␣
 ↪essential to further optimize SolarMax's performance, reduce production␣
 ↪costs, and address any potential environmental or safety concerns associated␣
 ↪with its use.
3. **Integration and Infrastructure:** The integration of SolarMax into␣
 ↪existing energy infrastructure and regulatory frameworks will require␣
 ↪careful planning and coordination to maximize its benefits and ensure a␣
 ↪smooth transition.
4. **Global Cooperation:** Achieving the full potential of SolarMax and other␣
 ↪renewable energy technologies will require international cooperation and␣
 ↪collaboration to overcome political, economic, and logistical challenges.
```

```
**Conclusion:**
The development of SolarMax represents a significant milestone in the pursuit␣
 ↪of sustainable energy solutions. With its unprecedented efficiency and␣
 ↪versatility, this breakthrough technology holds the promise of a cleaner,␣
 ↪more resilient energy future for generations to come.
 As researchers continue to push the boundaries of innovation and technology,␣
 ↪SolarMax serves as a shining example of human ingenuity and determination in␣
 ↪the face of one of the greatest challenges of our time: combating climate␣
 ↪change and securing a sustainable planet for future generations.
"""
```

[13]:

```
summary = generate_summary(article)
print("Summary:")
print(summary)
```

```
Summary:
With its unprecedented efficiency and versatility, this breakthrough technology
holds the promise of a cleaner, more resilient energy future for generations to
come.  As researchers continue to push the boundaries of innovation and
technology, SolarMax serves as a shining example of human ingenuity and
determination in the face of one of the greatest challenges of our time:
combating climate change and securing a sustainable planet for future
generations. **Accelerated Transition to Renewable Energy:** By significantly
increasing the efficiency of solar energy production, SolarMax could expedite
the transition away from fossil fuels, reducing greenhouse gas emissions and
mitigating the effects of climate change. **Integration and Infrastructure:**
The integration of SolarMax into existing energy infrastructure and regulatory
frameworks will require careful planning and coordination to maximize its
benefits and ensure a smooth transition. **Economic Opportunities:** The
development and deployment of SolarMax could create new economic opportunities,
driving job growth in the renewable energy sector and stimulating innovation and
investment.
```

5 Using BERT

[14]:

```
import warnings

warnings.simplefilter(action='ignore', category=FutureWarning)
```

[15]:

```
bert_model = Summarizer()
bert_summary = ''.join(bert_model(Context, min_length=100))
print('Bert Summary:')
print(bert_summary)
```

```
/usr/local/lib/python3.10/dist-packages/huggingface_hub/utils/_token.py:88:
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 in your Google Colab
and restart your session.
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:   0%|          | 0.00/571 [00:00<?, ?B/s]
```

```
model.safetensors:   0%|          | 0.00/1.34G [00:00<?, ?B/s]
```

```
tokenizer_config.json:   0%|          | 0.00/48.0 [00:00<?, ?B/s]
```

```
vocab.txt:   0%|          | 0.00/232k [00:00<?, ?B/s]
```

```
tokenizer.json:   0%|          | 0.00/466k [00:00<?, ?B/s]
```

```
Bert Summary:
**Title: Breakthrough in Renewable Energy: Novel Solar Technology Promises Game-
Changing Efficiency**

**Introduction:**
In a world increasingly grappling with climate change, the quest for sustainable
and efficient energy sources has never been more critical. **Background:**
Solar energy has long been hailed as a promising solution to the world's energy
needs. Traditional solar panels have made significant strides in recent years,
but their efficiency and scalability still pose challenges. **The
Breakthrough:**
The breakthrough technology, dubbed "SolarMax," represents a significant leap
forward in solar energy efficiency. Cost:** Despite its impressive efficiency,
SolarMax's initial manufacturing and installation costs may present barriers to
widespread adoption, particularly in developing countries or regions with
limited resources. With its unprecedented efficiency and versatility, this
breakthrough technology holds the promise of a cleaner, more resilient energy
future for generations to come.
```

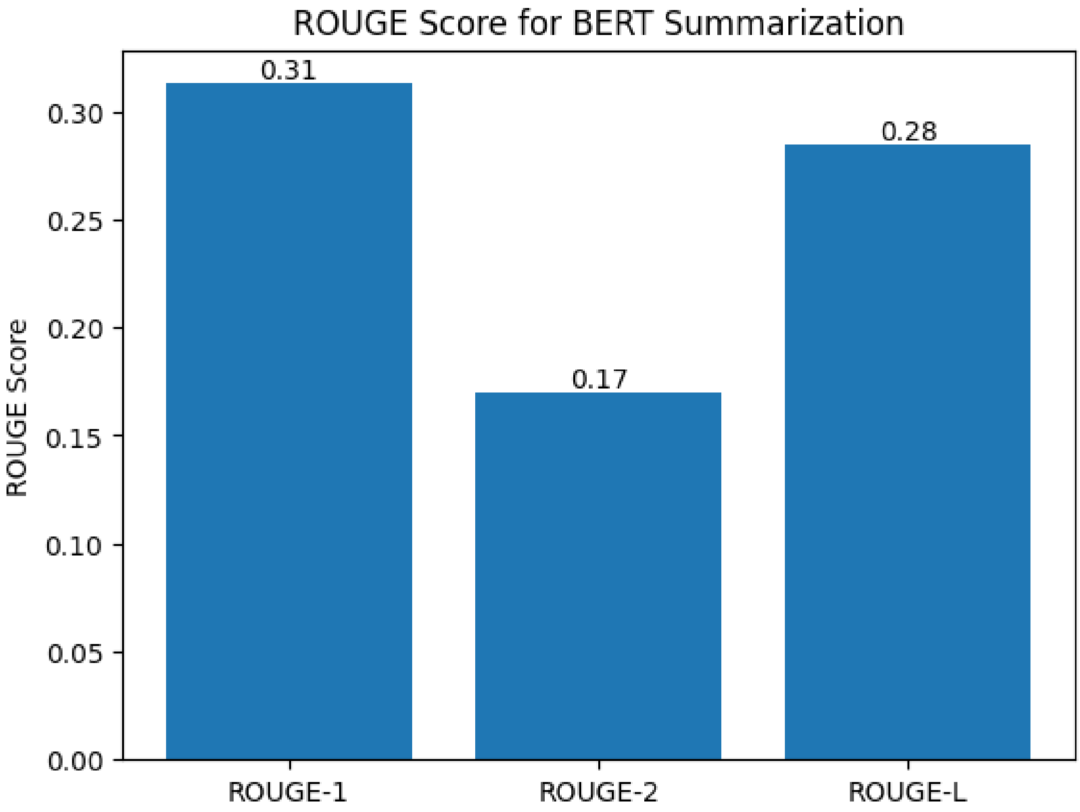
[16]:

```python
import matplotlib.pyplot as plt
import seaborn as sns
!pip install rouge
from rouge import Rouge

# Function to calculate ROUGE score
def calculate_rouge_score(bert_summary, summary):
    rouge = Rouge()
    scores = rouge.get_scores(bert_summary,summary)
    return scores[0]['rouge-1']['f'], scores[0]['rouge-2']['f'],␣
 ↪scores[0]['rouge-l']['f']
```

```
# Example data
bert_summary = '''
                  **Title: Breakthrough in Renewable Energy: Novel Solar␣
 ↪Technology Promises Game-Changing Efficiency**

**Introduction:**
In a world increasingly grappling with climate change, the quest for␣
 ↪sustainable and efficient energy sources has never been more critical.␣
 ↪**Background:**
Solar energy has long been hailed as a promising solution to the world's energy␣
 ↪needs. Traditional solar panels have made significant strides in recent␣
 ↪years, but their efficiency and scalability still pose challenges. **The␣
 ↪Breakthrough:**
The breakthrough technology, dubbed "SolarMax," represents a significant leap␣
 ↪forward in solar energy efficiency. Cost:** Despite its impressive␣
 ↪efficiency, SolarMax's initial manufacturing and installation costs may␣
 ↪present barriers to widespread adoption, particularly in developing␣
 ↪countries or regions with limited resources. With its unprecedented␣
 ↪efficiency and versatility, this breakthrough technology holds the promise␣
 ↪of a cleaner, more resilient energy future for generations to come.
                '''

actual_summary = '''
With its unprecedented efficiency and versatility, this breakthrough technology␣
 ↪holds the promise of a cleaner, more resilient energy future for generations␣
 ↪to come.  As researchers continue to push the boundaries of innovation and␣
 ↪technology, SolarMax serves as a shining example of human ingenuity and␣
 ↪determination in the face of one of the greatest challenges of our time:␣
 ↪combating climate change and securing a sustainable planet for future␣
 ↪generations. **Accelerated Transition to Renewable Energy:** By␣
 ↪significantly increasing the efficiency of solar energy production, SolarMax␣
 ↪could expedite the transition away from fossil fuels, reducing greenhouse␣
 ↪gas emissions and mitigating the effects of climate change. **Integration␣
 ↪and Infrastructure:** The integration of SolarMax into existing energy␣
 ↪infrastructure and regulatory frameworks will require careful planning and␣
 ↪coordination to maximize its benefits and ensure a smooth transition.␣
 ↪**Economic Opportunities:** The development and deployment of SolarMax could␣
 ↪create new economic opportunities, driving job growth in the renewable␣
 ↪energy sector and stimulating innovation and investment.
                  '''

# Calculate ROUGE score for BERT-generated summary
rouge_scores = calculate_rouge_score(bert_summary, summary)

# Plotting
labels = ['ROUGE-1', 'ROUGE-2', 'ROUGE-L']
scores = rouge_scores
```

```python
plt.bar(labels, scores)
plt.ylabel('ROUGE Score')
plt.title('ROUGE Score for BERT Summarization')

for i in range(len(scores)):
    plt.text(i, scores[i], f'{scores[i]:.2f}', ha='center', va='bottom')

plt.show()
```

```
Collecting rouge
  Downloading rouge-1.0.1-py3-none-any.whl (13 kB)
Requirement already satisfied: six in /usr/local/lib/python3.10/dist-packages
(from rouge) (1.16.0)
Installing collected packages: rouge
Successfully installed rouge-1.0.1
```



6 Using GPT-2

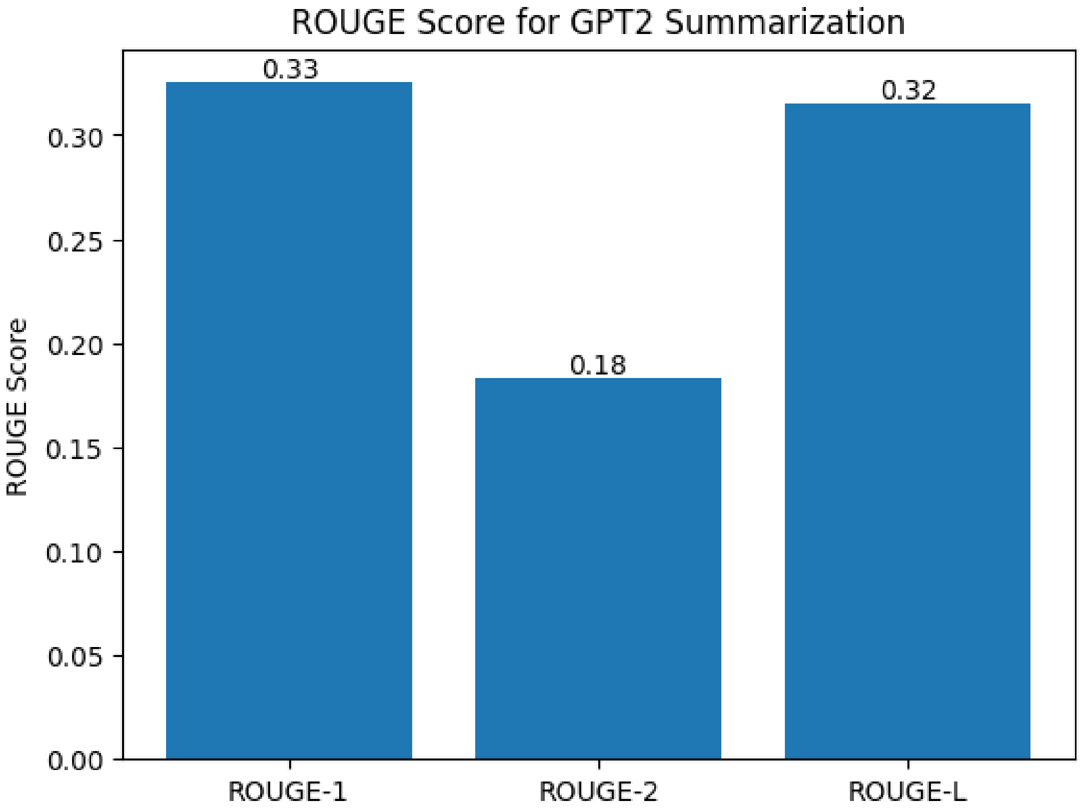
[17]:
```
gpt2_model = TransformerSummarizer(transformer_type = 'GPT2',␣
 ↪transformer_model_key = 'gpt2-medium')
gpt2_summary = ''.join(gpt2_model(Context, min_length=100))
print('gpt2 summary:', gpt2_summary)
```

```
config.json:   0%|          | 0.00/718 [00:00<?, ?B/s]

model.safetensors:   0%|          | 0.00/1.52G [00:00<?, ?B/s]

tokenizer_config.json:   0%|          | 0.00/26.0 [00:00<?, ?B/s]

vocab.json:   0%|          | 0.00/1.04M [00:00<?, ?B/s]

merges.txt:   0%|          | 0.00/456k [00:00<?, ?B/s]

tokenizer.json:   0%|          | 0.00/1.36M [00:00<?, ?B/s]

gpt2 summary: **Title: Breakthrough in Renewable Energy: Novel Solar Technology
Promises Game-Changing Efficiency**

**Introduction:**
In a world increasingly grappling with climate change, the quest for sustainable
and efficient energy sources has never been more critical. At the heart of
SolarMax lies a sophisticated nanomaterial that exhibits exceptional light-
absorbing properties. Energy Independence:** SolarMax's widespread adoption
could enhance energy independence for countries, reducing reliance on imported
fossil fuels and strengthening energy security. Cost:** Despite its impressive
efficiency, SolarMax's initial manufacturing and installation costs may present
barriers to widespread adoption, particularly in developing countries or regions
with limited resources. **Conclusion:**
The development of SolarMax represents a significant milestone in the pursuit of
sustainable energy solutions. With its unprecedented efficiency and versatility,
this breakthrough technology holds the promise of a cleaner, more resilient
energy future for generations to come.
```

[18]:
```
import matplotlib.pyplot as plt
import seaborn as sns
from rouge import Rouge

# Function to calculate ROUGE score
def calculate_rouge_score(gpt2_summary, summary):
    rouge = Rouge()
    scores = rouge.get_scores(gpt2_summary, summary)
    return scores[0]['rouge-1']['f'], scores[0]['rouge-2']['f'],␣
 ↪scores[0]['rouge-l']['f']

# Example data
gpt2_summary = '''
```

```
              **Introduction:**
In a world increasingly grappling with climate change, the quest for␣
 ↪sustainable and efficient energy sources has never been more critical. At␣
 ↪the heart of SolarMax lies a sophisticated nanomaterial that exhibits␣
 ↪exceptional light-absorbing properties. Energy Independence:** SolarMax's␣
 ↪widespread adoption could enhance energy independence for countries,␣
 ↪reducing reliance on imported fossil fuels and strengthening energy security.
 ↪ Cost:** Despite its impressive efficiency, SolarMax's initial manufacturing␣
 ↪and installation costs may present barriers to widespread adoption,␣
 ↪particularly in developing countries or regions with limited resources.␣
 ↪**Conclusion:**
The development of SolarMax represents a significant milestone in the pursuit␣
 ↪of sustainable energy solutions. With its unprecedented efficiency and␣
 ↪versatility, this breakthrough technology holds the promise of a cleaner,␣
 ↪more resilient energy future for generations to come.
              '''
actual_summary = '''
                  With its unprecedented efficiency and versatility, this␣
 ↪breakthrough technology holds the promise of a cleaner, more resilient␣
 ↪energy future for generations to come.
                   As researchers continue to push the boundaries of innovation␣
 ↪and technology, SolarMax serves as a shining example of human ingenuity and␣
 ↪determination in the face of one of the greatest challenges of our time:␣
 ↪combating climate change and securing a sustainable planet for future␣
 ↪generations. **Accelerated Transition to Renewable Energy:** By␣
 ↪significantly increasing the efficiency of solar energy production, SolarMax␣
 ↪could expedite the transition away from fossil fuels, reducing greenhouse␣
 ↪gas emissions and mitigating the effects of climate change. **Integration␣
 ↪and Infrastructure:** The integration of SolarMax into existing energy␣
 ↪infrastructure and regulatory frameworks will require careful planning and␣
 ↪coordination to maximize its benefits and ensure a smooth transition.
                  **Economic Opportunities:** The development and deployment of␣
 ↪SolarMax could create new economic opportunities, driving job growth in the␣
 ↪renewable energy sector and stimulating innovation and investment.
                  '''

# Calculate ROUGE score for BERT-generated summary
rouge_scores = calculate_rouge_score(gpt2_summary, summary)

# Plotting
labels = ['ROUGE-1', 'ROUGE-2', 'ROUGE-L']
scores = rouge_scores

plt.bar(labels, scores)
plt.ylabel('ROUGE Score')
plt.title('ROUGE Score for GPT2 Summarization')
```

```
for i in range(len(scores)):
    plt.text(i, scores[i], f'{scores[i]:.2f}', ha='center', va='bottom')

plt.show()
```



7 Using XL-NET

[19]:
```
XLNet_model = TransformerSummarizer(transformer_type = 'XLNet',␣
 ↪transformer_model_key = 'xlnet/xlnet-base-cased')
XLNet_summary = ''.join(XLNet_model(Context, min_length=100))
print('XLNet Summary:', XLNet_summary)
```

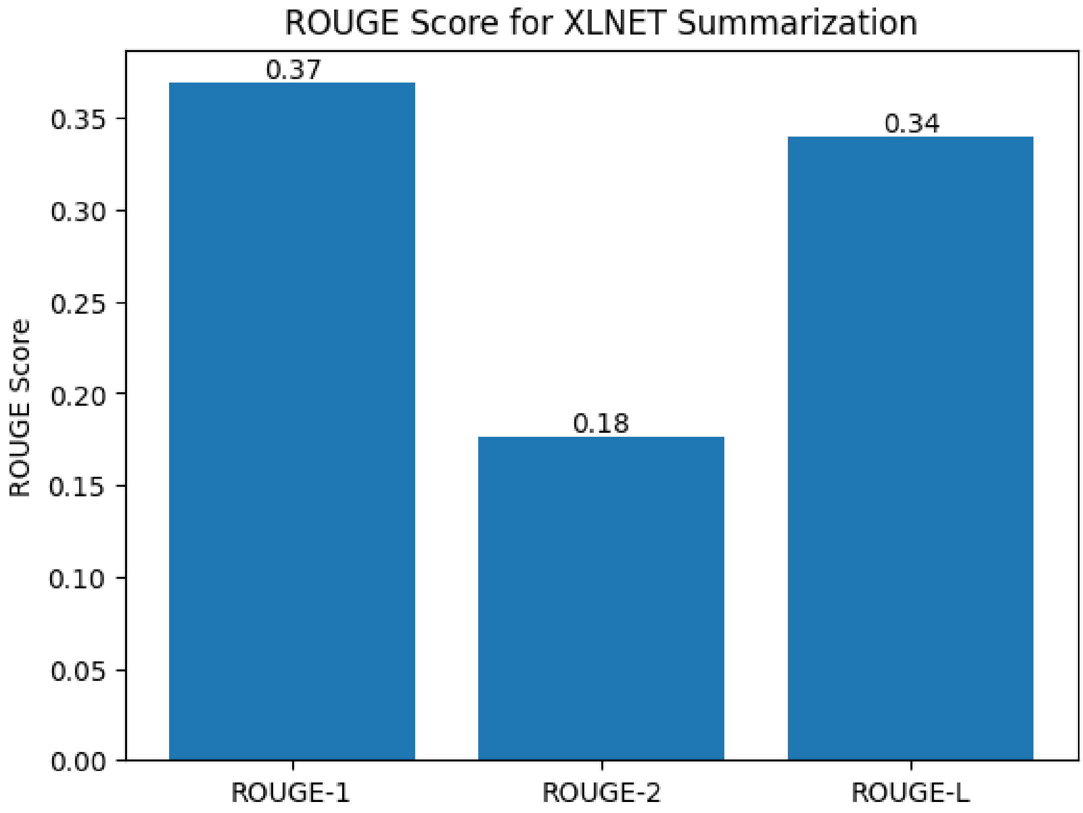
```
config.json:   0%|          | 0.00/760 [00:00<?, ?B/s]

pytorch_model.bin:   0%|          | 0.00/467M [00:00<?, ?B/s]

spiece.model:   0%|          | 0.00/798k [00:00<?, ?B/s]

tokenizer.json:   0%|          | 0.00/1.38M [00:00<?, ?B/s]

XLNet Summary: **Title: Breakthrough in Renewable Energy: Novel Solar Technology
Promises Game-Changing Efficiency**
```

```
**Introduction:**
In a world increasingly grappling with climate change, the quest for sustainable
and efficient energy sources has never been more critical. The latest
breakthrough in renewable energy comes in the form of revolutionary solar
technology that promises to transform the landscape of clean energy production.
Enhanced Efficiency:** SolarMax's groundbreaking design enables it to convert a
higher percentage of sunlight into electricity, maximizing energy output and
optimizing space utilization. Scalability:** The modular design of SolarMax
facilitates easy scalability, allowing for seamless integration into existing
solar infrastructure or the development of new solar projects of any size.
Economic Opportunities:** The development and deployment of SolarMax could
create new economic opportunities, driving job growth in the renewable energy
sector and stimulating innovation and investment. **Conclusion:**
The development of SolarMax represents a significant milestone in the pursuit of
sustainable energy solutions.
```

[20]:

```python
import matplotlib.pyplot as plt
import seaborn as sns
from rouge import Rouge

# Function to calculate ROUGE score
def calculate_rouge_score(XLNetsummary, summary):
    rouge = Rouge()
    scores = rouge.get_scores(XLNet_summary, summary)
    return scores[0]['rouge-1']['f'], scores[0]['rouge-2']['f'],␣
 ↪scores[0]['rouge-l']['f']

# Example data
XLNet_summary = '''
                **Title: Breakthrough in Renewable Energy: Novel Solar Technology␣
 ↪Promises Game-Changing Efficiency**

**Introduction:**
In a world increasingly grappling with climate change, the quest for␣
 ↪sustainable and efficient energy sources has never been more critical. The␣
 ↪latest breakthrough in renewable energy comes in the form of revolutionary␣
 ↪solar technology that promises to transform the landscape of clean energy␣
 ↪production. At the heart of SolarMax lies a sophisticated nanomaterial that␣
 ↪exhibits exceptional light-absorbing properties. Scalability:** The modular␣
 ↪design of SolarMax facilitates easy scalability, allowing for seamless␣
 ↪integration into existing solar infrastructure or the development of new␣
 ↪solar projects of any size. Economic Opportunities:** The development and␣
 ↪deployment of SolarMax could create new economic opportunities, driving job␣
 ↪growth in the renewable energy sector and stimulating innovation and␣
 ↪investment. **Conclusion:**
```

```
The development of SolarMax represents a significant milestone in the pursuit of sustainable energy solutions.
              '''
actual_summary = '''
                  With its unprecedented efficiency and versatility, this breakthrough technology holds the promise of a cleaner, more resilient energy future for generations to come.
                   As researchers continue to push the boundaries of innovation and technology, SolarMax serves as a shining example of human ingenuity and determination in the face of one of the greatest challenges of our time: combating climate change and securing a sustainable planet for future generations. **Accelerated Transition to Renewable Energy:** By significantly increasing the efficiency of solar energy production, SolarMax could expedite the transition away from fossil fuels, reducing greenhouse gas emissions and mitigating the effects of climate change. **Integration and Infrastructure:** The integration of SolarMax into existing energy infrastructure and regulatory frameworks will require careful planning and coordination to maximize its benefits and ensure a smooth transition.
                  **Economic Opportunities:** The development and deployment of SolarMax could create new economic opportunities, driving job growth in the renewable energy sector and stimulating innovation and investment.
                  '''

# Calculate ROUGE score for BERT-generated summary
rouge_scores = calculate_rouge_score(XLNet_summary, summary)

# Plotting
labels = ['ROUGE-1', 'ROUGE-2', 'ROUGE-L']
scores = rouge_scores

plt.bar(labels, scores)
plt.ylabel('ROUGE Score')
plt.title('ROUGE Score for XLNET Summarization')

for i in range(len(scores)):
    plt.text(i, scores[i], f'{scores[i]:.2f}', ha='center', va='bottom')

plt.show()
```



[21]:
```
!pip install transformers
```

```
Requirement already satisfied: transformers in /usr/local/lib/python3.10/dist-
packages (4.38.2)
Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-
packages (from transformers) (3.13.4)
Requirement already satisfied: huggingface-hub<1.0,>=0.19.3 in
/usr/local/lib/python3.10/dist-packages (from transformers) (0.20.3)
Requirement already satisfied: numpy>=1.17 in /usr/local/lib/python3.10/dist-
packages (from transformers) (1.25.2)
Requirement already satisfied: packaging>=20.0 in
/usr/local/lib/python3.10/dist-packages (from transformers) (24.0)
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.10/dist-
packages (from transformers) (6.0.1)
Requirement already satisfied: regex!=2019.12.17 in
/usr/local/lib/python3.10/dist-packages (from transformers) (2023.12.25)
Requirement already satisfied: requests in /usr/local/lib/python3.10/dist-
packages (from transformers) (2.31.0)
Requirement already satisfied: tokenizers<0.19,>=0.14 in
/usr/local/lib/python3.10/dist-packages (from transformers) (0.15.2)
Requirement already satisfied: safetensors>=0.4.1 in
```

```
/usr/local/lib/python3.10/dist-packages (from transformers) (0.4.3)
Requirement already satisfied: tqdm>=4.27 in /usr/local/lib/python3.10/dist-
packages (from transformers) (4.66.2)
Requirement already satisfied: fsspec>=2023.5.0 in
/usr/local/lib/python3.10/dist-packages (from huggingface-
hub<1.0,>=0.19.3->transformers) (2023.6.0)
Requirement already satisfied: typing-extensions>=3.7.4.3 in
/usr/local/lib/python3.10/dist-packages (from huggingface-
hub<1.0,>=0.19.3->transformers) (4.11.0)
Requirement already satisfied: charset-normalizer<4,>=2 in
/usr/local/lib/python3.10/dist-packages (from requests->transformers) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-
packages (from requests->transformers) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/usr/local/lib/python3.10/dist-packages (from requests->transformers) (2.0.7)
Requirement already satisfied: certifi>=2017.4.17 in
/usr/local/lib/python3.10/dist-packages (from requests->transformers) (2024.2.2)
```

[22]:

```python
import warnings

warnings.simplefilter(action='ignore', category=FutureWarning)
```

8 Using PEGASUS

[23]:

```python
from transformers import PegasusForConditionalGeneration, PegasusTokenizer

pegasus_tokenizer = PegasusTokenizer.from_pretrained('google/pegasus-large')
pegasus_model = PegasusForConditionalGeneration.from_pretrained("google/
  ↪pegasus-large")

inputs = pegasus_tokenizer([Context], max_length = 900, return_tensors='pt',␣
  ↪truncation=True)

pegasus_summary_ids = pegasus_model.generate(inputs['input_ids'], max_length␣
  ↪=200, min_length=60, length_penalty=2.0, num_beams=4, early_stopping=True)
pegasus_summary = pegasus_tokenizer.decode(pegasus_summary_ids[0],␣
  ↪skip_special_tokens=True)

print('Pegasus Summary:', pegasus_summary)
```

```
tokenizer_config.json:   0%|          | 0.00/88.0 [00:00<?, ?B/s]

spiece.model:   0%|          | 0.00/1.91M [00:00<?, ?B/s]

special_tokens_map.json:   0%|          | 0.00/65.0 [00:00<?, ?B/s]

config.json:   0%|          | 0.00/3.09k [00:00<?, ?B/s]
```

```
pytorch_model.bin:   0%|          | 0.00/2.28G [00:00<?, ?B/s]
```

```
Some weights of PegasusForConditionalGeneration were not initialized from the
model checkpoint at google/pegasus-large and are newly initialized:
['model.decoder.embed_positions.weight', 'model.encoder.embed_positions.weight']
You should probably TRAIN this model on a down-stream task to be able to use it
for predictions and inference.
```

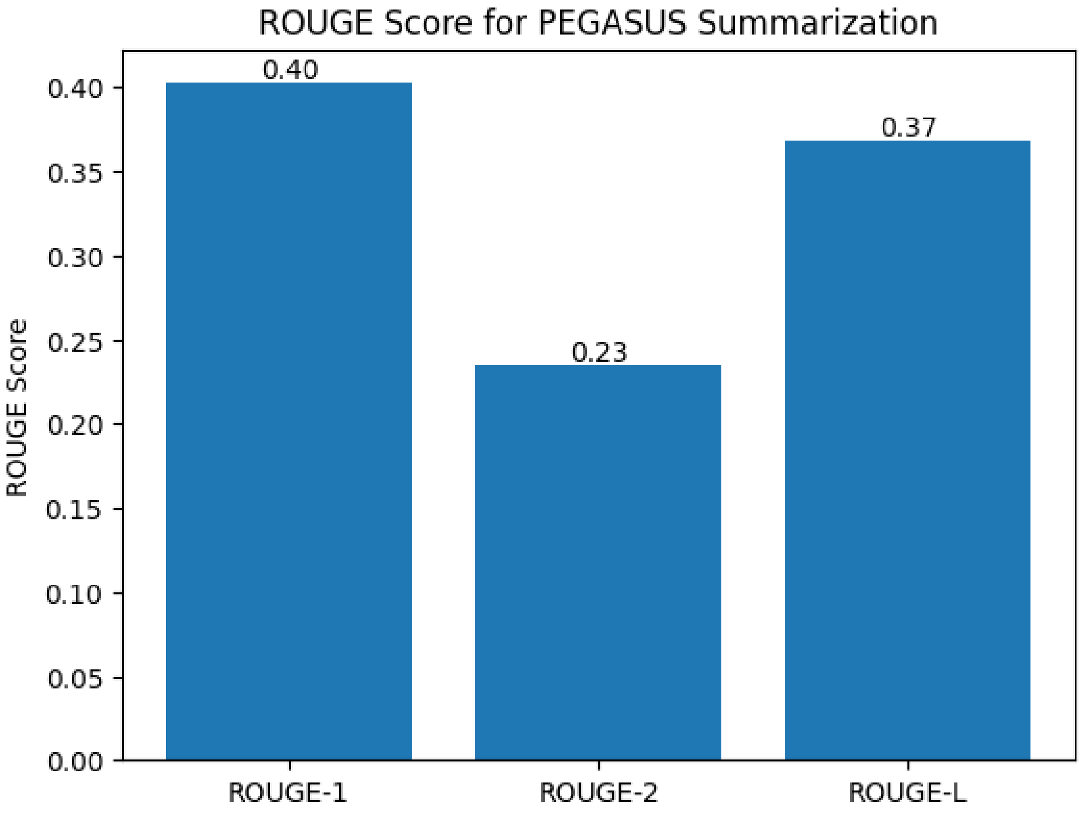
```
generation_config.json:   0%|          | 0.00/260 [00:00<?, ?B/s]
```

```
Pegasus Summary: **Title: Breakthrough in Renewable Energy: Novel Solar
Technology Promises Game-Changing Efficiency** **Introduction:** In a world
increasingly grappling with climate change, the quest for sustainable and
efficient energy sources has never been more critical. With its unparalleled
efficiency and versatility, this breakthrough technology has the potential to
revolutionize the global energy landscape in several ways: 1. **Accelerated
Transition to Renewable Energy:** By significantly increasing the efficiency of
solar energy production, SolarMax could expedite the transition away from fossil
fuels.
```

[24]:

```python
import matplotlib.pyplot as plt
import seaborn as sns
from rouge import Rouge

# Function to calculate ROUGE score
def calculate_rouge_score(pegasus_summary, summary):
    rouge = Rouge()
    scores = rouge.get_scores(pegasus_summary, summary)
    return scores[0]['rouge-1']['f'], scores[0]['rouge-2']['f'],␣
 ↪scores[0]['rouge-l']['f']

# Example data
pegasus_summary = '''
              **Title: Breakthrough in Renewable Energy: Novel Solar Technology␣
 ↪Promises Game-Changing Efficiency**
              **Introduction:** In a world increasingly grappling with climate␣
 ↪change, the quest for sustainable and efficient energy sources has never␣
 ↪been more critical. With its unparalleled efficiency and versatility, this␣
 ↪breakthrough technology has the potential to revolutionize the global energy␣
 ↪landscape in several ways: 1. **Accelerated Transition to Renewable Energy:
 ↪** By significantly increasing the efficiency of solar energy production,␣
 ↪SolarMax could expedite the transition away from fossil fuels.
              '''
actual_summary = '''
                  With its unprecedented efficiency and versatility, this␣
 ↪breakthrough technology holds the promise of a cleaner, more resilient␣
 ↪energy future for generations to come.
```

```
                   As researchers continue to push the boundaries of innovation␣
 ↪and technology, SolarMax serves as a shining example of human ingenuity and␣
 ↪determination in the face of one of the greatest challenges of our time:␣
 ↪combating climate change and securing a sustainable planet for future␣
 ↪generations. **Accelerated Transition to Renewable Energy:** By␣
 ↪significantly increasing the efficiency of solar energy production, SolarMax␣
 ↪could expedite the transition away from fossil fuels, reducing greenhouse␣
 ↪gas emissions and mitigating the effects of climate change. **Integration␣
 ↪and Infrastructure:** The integration of SolarMax into existing energy␣
 ↪infrastructure and regulatory frameworks will require careful planning and␣
 ↪coordination to maximize its benefits and ensure a smooth transition.
                  **Economic Opportunities:** The development and deployment of␣
 ↪SolarMax could create new economic opportunities, driving job growth in the␣
 ↪renewable energy sector and stimulating innovation and investment.
                  '''

# Calculate ROUGE score for BERT-generated summary
rouge_scores = calculate_rouge_score(pegasus_summary, summary)

# Plotting
labels = ['ROUGE-1', 'ROUGE-2', 'ROUGE-L']
scores = rouge_scores

plt.bar(labels, scores)
plt.ylabel('ROUGE Score')
plt.title('ROUGE Score for PEGASUS Summarization')

for i in range(len(scores)):
    plt.text(i, scores[i], f'{scores[i]:.2f}', ha='center', va='bottom')

plt.show()
```



8.1 Using T5 (Transfer text-to-text Transformer)

[25]:
```
from transformers import T5ForConditionalGeneration, T5Tokenizer

# T5
t5_tokenizer = T5Tokenizer.from_pretrained("t5-large")
t5_model = T5ForConditionalGeneration.from_pretrained("t5-large")
inputs = t5_tokenizer([Context], max_length=1024, return_tensors='pt',␣
 ↪truncation=True)
t5_summary_ids = t5_model.generate(inputs['input_ids'], max_length=150,␣
 ↪min_length=40, length_penalty=2.0, num_beams=4, early_stopping=True)
t5_summary = t5_tokenizer.decode(t5_summary_ids[0], skip_special_tokens=True)
print("T5 Summary:")
print(t5_summary)
```

```
spiece.model:   0%|          | 0.00/792k [00:00<?, ?B/s]

tokenizer.json:   0%|          | 0.00/1.39M [00:00<?, ?B/s]

config.json:   0%|          | 0.00/1.21k [00:00<?, ?B/s]

You are using the default legacy behaviour of the <class
```

```
'transformers.models.t5.tokenization_t5.T5Tokenizer'>. This is expected, and
simply means that the `legacy` (previous) behavior will be used so nothing
changes for you. If you want to use the new behaviour, set `legacy=False`. This
should only be set if you understand what it means, and thoroughly read the
reason why this was added as explained in
https://github.com/huggingface/transformers/pull/24565
Special tokens have been added in the vocabulary, make sure the associated word
embeddings are fine-tuned or trained.
```

```
model.safetensors:   0%|          | 0.00/2.95G [00:00<?, ?B/s]
```

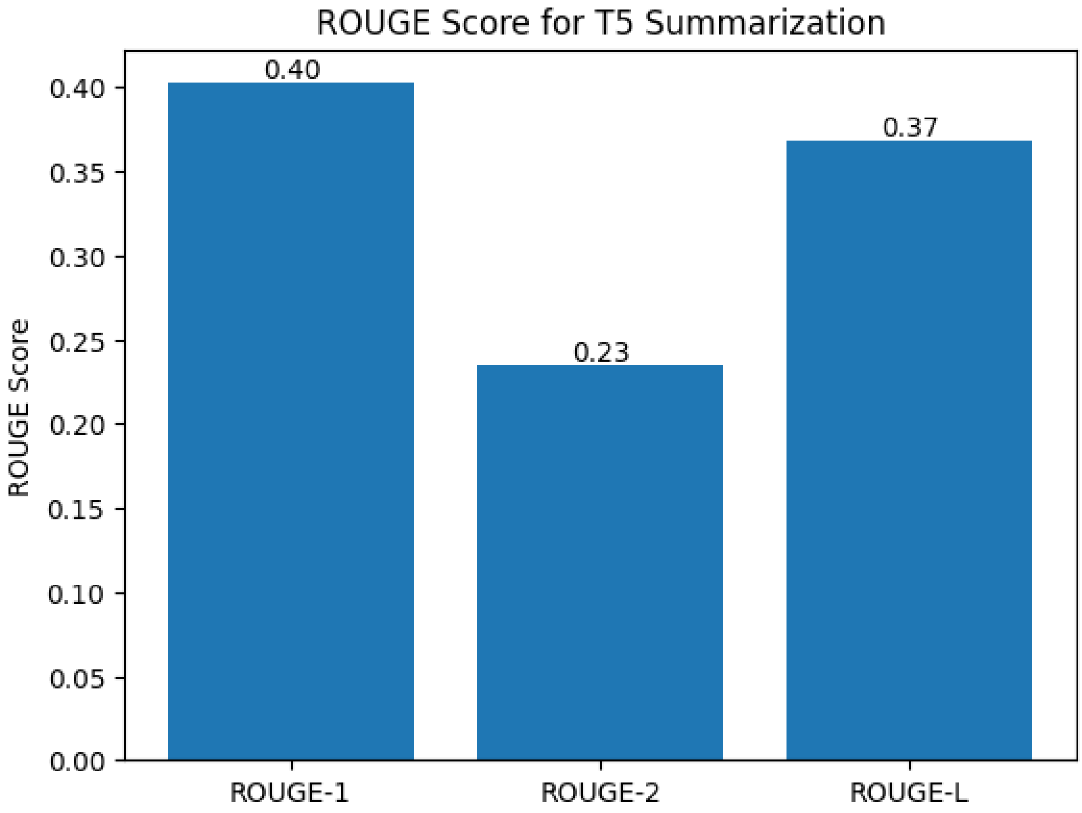
```
generation_config.json:   0%|          | 0.00/147 [00:00<?, ?B/s]
```

```
T5 Summary:
. SolarMax represents a significant breakthrough in renewable energy technology
that holds the promise of a cleaner, more resilient energy future for future
generations. SolarMax is a shining example of human determination and
determination in the face of one of the greatest challenges of our time:
combating climate change and securing a sustainable planet for future
generations. **About SolarMax:** SolarMax was developed by a team of researchers
at [Institution/Company]. **About SolarMax:** SolarMax was developed by
```

[26]:

```python
import matplotlib.pyplot as plt
import seaborn as sns
from rouge import Rouge

# Function to calculate ROUGE score
def calculate_rouge_score(t5_summary, summary):
    rouge = Rouge()
    scores = rouge.get_scores(t5_summary, summary)
    return scores[0]['rouge-1']['f'], scores[0]['rouge-2']['f'],␣
 ↪scores[0]['rouge-l']['f']

# Example data
t5_summary = '''
              **Title: Breakthrough in Renewable Energy: Novel Solar Technology␣
 ↪Promises Game-Changing Efficiency**
              **Introduction:** In a world increasingly grappling with climate␣
 ↪change, the quest for sustainable and efficient energy sources has never␣
 ↪been more critical. With its unparalleled efficiency and versatility, this␣
 ↪breakthrough technology has the potential to revolutionize the global energy␣
 ↪landscape in several ways: 1. **Accelerated Transition to Renewable Energy:
 ↪** By significantly increasing the efficiency of solar energy production,␣
 ↪SolarMax could expedite the transition away from fossil fuels.
              '''
actual_summary = '''
                  With its unprecedented efficiency and versatility, this␣
 ↪breakthrough technology holds the promise of a cleaner, more resilient␣
 ↪energy future for generations to come.
```

```
                   As researchers continue to push the boundaries of innovation␣
 ↪and technology, SolarMax serves as a shining example of human ingenuity and␣
 ↪determination in the face of one of the greatest challenges of our time:␣
 ↪combating climate change and securing a sustainable planet for future␣
 ↪generations. **Accelerated Transition to Renewable Energy:** By␣
 ↪significantly increasing the efficiency of solar energy production, SolarMax␣
 ↪could expedite the transition away from fossil fuels, reducing greenhouse␣
 ↪gas emissions and mitigating the effects of climate change. **Integration␣
 ↪and Infrastructure:** The integration of SolarMax into existing energy␣
 ↪infrastructure and regulatory frameworks will require careful planning and␣
 ↪coordination to maximize its benefits and ensure a smooth transition.
                  **Economic Opportunities:** The development and deployment of␣
 ↪SolarMax could create new economic opportunities, driving job growth in the␣
 ↪renewable energy sector and stimulating innovation and investment.
                  '''

# Calculate ROUGE score for BERT-generated summary
rouge_scores = calculate_rouge_score(t5_summary, summary)

# Plotting
labels = ['ROUGE-1', 'ROUGE-2', 'ROUGE-L']
scores = rouge_scores

plt.bar(labels, scores)
plt.ylabel('ROUGE Score')
plt.title('ROUGE Score for T5 Summarization')

for i in range(len(scores)):
    plt.text(i, scores[i], f'{scores[i]:.2f}', ha='center', va='bottom')

plt.show()
```



Next_Word Prediction(Text Generation)

```
[28]: print(Context)
```

```
**Title: Breakthrough in Renewable Energy: Novel Solar Technology Promises Game-
Changing Efficiency**

**Introduction:**
In a world increasingly grappling with climate change, the quest for sustainable
and efficient energy sources has never been more critical. The latest
breakthrough in renewable energy comes in the form of revolutionary solar
technology that promises to transform the landscape of clean energy production.
Developed by a team of researchers at [Institution/Company], this innovation
boasts unprecedented efficiency levels that could potentially revolutionize the
global energy sector.

**Background:**
Solar energy has long been hailed as a promising solution to the world's energy
needs. Traditional solar panels have made significant strides in recent years,
but their efficiency and scalability still pose challenges. However, the
emergence of next-generation solar technologies offers renewed hope for a future
```

powered by clean, renewable energy.

The Breakthrough:
The breakthrough technology, dubbed "SolarMax," represents a significant leap forward in solar energy efficiency. Unlike conventional solar panels that rely on photovoltaic cells to convert sunlight into electricity, SolarMax utilizes a novel approach inspired by [describe the inspiration/source of innovation, e.g., biological processes or quantum mechanics].

At the heart of SolarMax lies a sophisticated nanomaterial that exhibits exceptional light-absorbing properties. This material, developed through years of research and experimentation, allows SolarMax to capture a broader spectrum of sunlight, including wavelengths that traditional solar panels cannot harness effectively. As a result, SolarMax achieves an unprecedented efficiency rating of over 40%, far surpassing the capabilities of current solar technologies.

Key Features and Advantages:
1. **Enhanced Efficiency:** SolarMax's groundbreaking design enables it to convert a higher percentage of sunlight into electricity, maximizing energy output and optimizing space utilization.
2. **Versatility:** SolarMax is adaptable to various environments and applications, from residential rooftops to large-scale solar farms, making it a versatile solution for diverse energy needs.
3. **Durability:** The nanomaterial used in SolarMax is highly durable and resilient, ensuring long-term performance and minimal maintenance requirements.
4. **Scalability:** The modular design of SolarMax facilitates easy scalability, allowing for seamless integration into existing solar infrastructure or the development of new solar projects of any size.

Potential Impact:
The implications of SolarMax extend far beyond its impressive technical specifications. With its unparalleled efficiency and versatility, this breakthrough technology has the potential to revolutionize the global energy landscape in several ways:

1. **Accelerated Transition to Renewable Energy:** By significantly increasing the efficiency of solar energy production, SolarMax could expedite the transition away from fossil fuels, reducing greenhouse gas emissions and mitigating the effects of climate change.
2. **Energy Independence:** SolarMax's widespread adoption could enhance energy independence for countries, reducing reliance on imported fossil fuels and strengthening energy security.
3. **Economic Opportunities:** The development and deployment of SolarMax could create new economic opportunities, driving job growth in the renewable energy sector and stimulating innovation and investment.
4. **Access to Clean Energy:** SolarMax's scalability and affordability could improve access to clean energy in underserved communities, empowering individuals and regions with sustainable power sources.

```
**Challenges and Future Directions:**
While SolarMax represents a remarkable achievement in renewable energy
technology, several challenges and considerations remain:

1. **Cost:** Despite its impressive efficiency, SolarMax's initial manufacturing
and installation costs may present barriers to widespread adoption, particularly
in developing countries or regions with limited resources.
2. **Technological Advancements:** Continued research and development are
essential to further optimize SolarMax's performance, reduce production costs,
and address any potential environmental or safety concerns associated with its
use.
3. **Integration and Infrastructure:** The integration of SolarMax into existing
energy infrastructure and regulatory frameworks will require careful planning
and coordination to maximize its benefits and ensure a smooth transition.
4. **Global Cooperation:** Achieving the full potential of SolarMax and other
renewable energy technologies will require international cooperation and
collaboration to overcome political, economic, and logistical challenges.

**Conclusion:**
The development of SolarMax represents a significant milestone in the pursuit of
sustainable energy solutions. With its unprecedented efficiency and versatility,
this breakthrough technology holds the promise of a cleaner, more resilient
energy future for generations to come. As researchers continue to push the
boundaries of innovation and technology, SolarMax serves as a shining example of
human ingenuity and determination in the face of one of the greatest challenges
of our time: combating climate change and securing a sustainable planet for
future generations.
```

[29]:

```
import tensorflow as tf
from tensorflow.keras.preprocessing.text import Tokenizer
```

[30]:

```
tokenizer = Tokenizer()
```

[32]:

```
tokenizer.fit_on_texts([Context])
```

[33]:

```
len(tokenizer.word_index)  # Number of words.
```

[33]:

```
343
```

[34]:

```
tokenizer.word_index   # Tokanizing integer to the each word in the text.
```

[34]:

```
{'and': 1,
 'the': 2,
 'of': 3,
 'energy': 4,
 'a': 5,
 'to': 6,
 'solarmax': 7,
 'in': 8,
 'solar': 9,
 'efficiency': 10,
 'renewable': 11,
 'technology': 12,
 'for': 13,
 'breakthrough': 14,
 'with': 15,
 'its': 16,
 'that': 17,
 'could': 18,
 'challenges': 19,
 "solarmax's": 20,
 'sustainable': 21,
 'clean': 22,
 'by': 23,
 'this': 24,
 'innovation': 25,
 'as': 26,
 'scalability': 27,
 'future': 28,
 'into': 29,
 'or': 30,
 'development': 31,
 'potential': 32,
 'climate': 33,
 'change': 34,
 'has': 35,
 'production': 36,
 'unprecedented': 37,
 'global': 38,
 'panels': 39,
 'significant': 40,
 'technologies': 41,
 'represents': 42,
 'sunlight': 43,
 '1': 44,
 '2': 45,
 'versatility': 46,
 '3': 47,
 '4': 48,
 'integration': 49,
 'infrastructure': 50,
 'transition': 51,
 'economic': 52,
 'novel': 53,
 'promises': 54,
 'sources': 55,
 'been': 56,
 'more': 57,
 'landscape': 58,
 'developed': 59,
 'researchers': 60,
 'at': 61,
 'revolutionize': 62,
 'sector': 63,
 'long': 64,
 'solution': 65,
 'needs': 66,
 'traditional': 67,
 'years': 68,
 'on': 69,
 'convert': 70,
 'electricity': 71,
 'nanomaterial': 72,
 'research': 73,
 'far': 74,
 'design': 75,
 'it': 76,
 'is': 77,
 'from': 78,
 'resilient': 79,
 'performance': 80,
 'existing': 81,
 'new': 82,
 'any': 83,
 'impressive': 84,
 'several': 85,
 'fossil': 86,
 'fuels': 87,
 'reducing': 88,
 'independence': 89,
 'widespread': 90,
 'adoption': 91,
 'countries': 92,
 'opportunities': 93,
 'access': 94,
 'regions': 95,
 'costs': 96,
 'will': 97,
 'require': 98,
 'cooperation': 99,
 'generations': 100,
 'title': 101,
 'game': 102,
 'changing': 103,
 'introduction': 104,
 'world': 105,
 'increasingly': 106,
 'grappling': 107,
 'quest': 108,
 'efficient': 109,
 'never': 110,
 'critical': 111,
 'latest': 112,
 'comes': 113,
 'form': 114,
 'revolutionary': 115,
 'transform': 116,
 'team': 117,
 'institution': 118,
 'company': 119,
 'boasts': 120,
 'levels': 121,
 'potentially': 122,
 'background': 123,
 'hailed': 124,
 'promising': 125,
 "world's": 126,
 'have': 127,
 'made': 128,
 'strides': 129,
 'recent': 130,
 'but': 131,
 'their': 132,
 'still': 133,
 'pose': 134,
 'however': 135,
 'emergence': 136,
 'next': 137,
 'generation': 138,
 'offers': 139,
 'renewed': 140,
 'hope': 141,
 'powered': 142,
 'dubbed': 143,
 'leap': 144,
 'forward': 145,
 'unlike': 146,
 'conventional': 147,
 'rely': 148,
 'photovoltaic': 149,
 'cells': 150,
 'utilizes': 151,
 'approach': 152,
 'inspired': 153,
 'describe': 154,
 'inspiration': 155,
 'source': 156,
 'e': 157,
 'g': 158,
 'biological': 159,
 'processes': 160,
 'quantum': 161,
 'mechanics': 162,
 'heart': 163,
 'lies': 164,
 'sophisticated': 165,
 'exhibits': 166,
 'exceptional': 167,
 'light': 168,
 'absorbing': 169,
 'properties': 170,
 'material': 171,
 'through': 172,
 'experimentation': 173,
 'allows': 174,
 'capture': 175,
 'broader': 176,
 'spectrum': 177,
 'including': 178,
 'wavelengths': 179,
 'cannot': 180,
 'harness': 181,
 'effectively': 182,
 'result': 183,
 'achieves': 184,
 'an': 185,
 'rating': 186,
 'over': 187,
 '40': 188,
 'surpassing': 189,
 'capabilities': 190,
 'current': 191,
 'key': 192,
 'features': 193,
 'advantages': 194,
 'enhanced': 195,
 'groundbreaking': 196,
 'enables': 197,
 'higher': 198,
 'percentage': 199,
 'maximizing': 200,
 'output': 201,
 'optimizing': 202,
 'space': 203,
 'utilization': 204,
 'adaptable': 205,
 'various': 206,
 'environments': 207,
 'applications': 208,
 'residential': 209,
 'rooftops': 210,
 'large': 211,
 'scale': 212,
 'farms': 213,
 'making': 214,
 'versatile': 215,
 'diverse': 216,
 'durability': 217,
 'used': 218,
 'highly': 219,
 'durable': 220,
 'ensuring': 221,
 'term': 222,
 'minimal': 223,
 'maintenance': 224,
 'requirements': 225,
 'modular': 226,
 'facilitates': 227,
 'easy': 228,
 'allowing': 229,
 'seamless': 230,
 'projects': 231,
 'size': 232,
 'impact': 233,
 'implications': 234,
 'extend': 235,
 'beyond': 236,
 'technical': 237,
 'specifications': 238,
 'unparalleled': 239,
 'ways': 240,
 'accelerated': 241,
 'significantly': 242,
 'increasing': 243,
 'expedite': 244,
 'away': 245,
 'greenhouse': 246,
 'gas': 247,
 'emissions': 248,
 'mitigating': 249,
 'effects': 250,
 'enhance': 251,
 'reliance': 252,
 'imported': 253,
 'strengthening': 254,
 'security': 255,
 'deployment': 256,
 'create': 257,
 'driving': 258,
 'job': 259,
 'growth': 260,
 'stimulating': 261,
 'investment': 262,
 'affordability': 263,
 'improve': 264,
 'underserved': 265,
 'communities': 266,
 'empowering': 267,
 'individuals': 268,
 'power': 269,
 'directions': 270,
 'while': 271,
 'remarkable': 272,
 'achievement': 273,
 'considerations': 274,
 'remain': 275,
 'cost': 276,
 'despite': 277,
 'initial': 278,
 'manufacturing': 279,
 'installation': 280,
 'may': 281,
 'present': 282,
 'barriers': 283,
 'particularly': 284,
 'developing': 285,
 'limited': 286,
 'resources': 287,
 'technological': 288,
 'advancements': 289,
 'continued': 290,
 'are': 291,
 'essential': 292,
 'further': 293,
 'optimize': 294,
 'reduce': 295,
 'address': 296,
 'environmental': 297,
 'safety': 298,
 'concerns': 299,
 'associated': 300,
 'use': 301,
 'regulatory': 302,
 'frameworks': 303,
 'careful': 304,
 'planning': 305,
 'coordination': 306,
 'maximize': 307,
 'benefits': 308,
 'ensure': 309,
 'smooth': 310,
 'achieving': 311,
 'full': 312,
 'other': 313,
 'international': 314,
 'collaboration': 315,
 'overcome': 316,
 'political': 317,
 'logistical': 318,
 'conclusion': 319,
 'milestone': 320,
 'pursuit': 321,
 'solutions': 322,
 'holds': 323,
 'promise': 324,
 'cleaner': 325,
 'come': 326,
 'continue': 327,
 'push': 328,
 'boundaries': 329,
 'serves': 330,
 'shining': 331,
 'example': 332,
 'human': 333,
 'ingenuity': 334,
 'determination': 335,
 'face': 336,
 'one': 337,
 'greatest': 338,
 'our': 339,
 'time': 340,
 'combating': 341,
 'securing': 342,
 'planet': 343}
```

[35]:

```
input_sequences = []
for sentence in Context.split('\n'):
  print(tokenizer.texts_to_sequences([sentence]))
# Here this showing it in 2D array. eg.-> [[93, 1, 13]]

# lets convert it.
```

```
[[101, 14, 8, 11, 4, 53, 9, 12, 54, 102, 103, 10]]
[[]]
[[104]]
[[8, 5, 105, 106, 107, 15, 33, 34, 2, 108, 13, 21, 1, 109, 4, 55, 35, 110, 56,
57, 111, 2, 112, 14, 8, 11, 4, 113, 8, 2, 114, 3, 115, 9, 12, 17, 54, 6, 116, 2,
58, 3, 22, 4, 36, 59, 23, 5, 117, 3, 60, 61, 118, 119, 24, 25, 120, 37, 10, 121,
17, 18, 122, 62, 2, 38, 4, 63]]
[[]]
[[123]]
[[9, 4, 35, 64, 56, 124, 26, 5, 125, 65, 6, 2, 126, 4, 66, 67, 9, 39, 127, 128,
40, 129, 8, 130, 68, 131, 132, 10, 1, 27, 133, 134, 19, 135, 2, 136, 3, 137,
138, 9, 41, 139, 140, 141, 13, 5, 28, 142, 23, 22, 11, 4]]
[[]]
[[2, 14]]
[[2, 14, 12, 143, 7, 42, 5, 40, 144, 145, 8, 9, 4, 10, 146, 147, 9, 39, 17, 148,
69, 149, 150, 6, 70, 43, 29, 71, 7, 151, 5, 53, 152, 153, 23, 154, 2, 155, 156,
3, 25, 157, 158, 159, 160, 30, 161, 162]]
[[]]
[[61, 2, 163, 3, 7, 164, 5, 165, 72, 17, 166, 167, 168, 169, 170, 24, 171, 59,
172, 68, 3, 73, 1, 173, 174, 7, 6, 175, 5, 176, 177, 3, 43, 178, 179, 17, 67, 9,
39, 180, 181, 182, 26, 5, 183, 7, 184, 185, 37, 10, 186, 3, 187, 188, 74, 189,
2, 190, 3, 191, 9, 41]]
[[]]
[[192, 193, 1, 194]]
[[44, 195, 10, 20, 196, 75, 197, 76, 6, 70, 5, 198, 199, 3, 43, 29, 71, 200, 4,
201, 1, 202, 203, 204]]
[[45, 46, 7, 77, 205, 6, 206, 207, 1, 208, 78, 209, 210, 6, 211, 212, 9, 213,
214, 76, 5, 215, 65, 13, 216, 4, 66]]
```

```
[[47, 217, 2, 72, 218, 8, 7, 77, 219, 220, 1, 79, 221, 64, 222, 80, 1, 223, 224,
225]]
[[48, 27, 2, 226, 75, 3, 7, 227, 228, 27, 229, 13, 230, 49, 29, 81, 9, 50, 30,
2, 31, 3, 82, 9, 231, 3, 83, 232]]
[[]]
[[32, 233]]
[[2, 234, 3, 7, 235, 74, 236, 16, 84, 237, 238, 15, 16, 239, 10, 1, 46, 24, 14,
12, 35, 2, 32, 6, 62, 2, 38, 4, 58, 8, 85, 240]]
[[]]
[[44, 241, 51, 6, 11, 4, 23, 242, 243, 2, 10, 3, 9, 4, 36, 7, 18, 244, 2, 51,
245, 78, 86, 87, 88, 246, 247, 248, 1, 249, 2, 250, 3, 33, 34]]
[[45, 4, 89, 20, 90, 91, 18, 251, 4, 89, 13, 92, 88, 252, 69, 253, 86, 87, 1,
254, 4, 255]]
[[47, 52, 93, 2, 31, 1, 256, 3, 7, 18, 257, 82, 52, 93, 258, 259, 260, 8, 2, 11,
4, 63, 1, 261, 25, 1, 262]]
[[48, 94, 6, 22, 4, 20, 27, 1, 263, 18, 264, 94, 6, 22, 4, 8, 265, 266, 267,
268, 1, 95, 15, 21, 269, 55]]
[[]]
[[19, 1, 28, 270]]
[[271, 7, 42, 5, 272, 273, 8, 11, 4, 12, 85, 19, 1, 274, 275]]
[[]]
[[44, 276, 277, 16, 84, 10, 20, 278, 279, 1, 280, 96, 281, 282, 283, 6, 90, 91,
284, 8, 285, 92, 30, 95, 15, 286, 287]]
[[45, 288, 289, 290, 73, 1, 31, 291, 292, 6, 293, 294, 20, 80, 295, 36, 96, 1,
296, 83, 32, 297, 30, 298, 299, 300, 15, 16, 301]]
[[47, 49, 1, 50, 2, 49, 3, 7, 29, 81, 4, 50, 1, 302, 303, 97, 98, 304, 305, 1,
306, 6, 307, 16, 308, 1, 309, 5, 310, 51]]
[[48, 38, 99, 311, 2, 312, 32, 3, 7, 1, 313, 11, 4, 41, 97, 98, 314, 99, 1, 315,
6, 316, 317, 52, 1, 318, 19]]
[[]]
[[319]]
[[2, 31, 3, 7, 42, 5, 40, 320, 8, 2, 321, 3, 21, 4, 322, 15, 16, 37, 10, 1, 46,
24, 14, 12, 323, 2, 324, 3, 5, 325, 57, 79, 4, 28, 13, 100, 6, 326, 26, 60, 327,
6, 328, 2, 329, 3, 25, 1, 12, 7, 330, 26, 5, 331, 332, 3, 333, 334, 1, 335, 8,
2, 336, 3, 337, 3, 2, 338, 19, 3, 339, 340, 341, 33, 34, 1, 342, 5, 21, 343, 13,
28, 100]]
[[]]
[[]]
```

[38]:

```
input_sequences = []
for sentence in Context.split('\n'):
  tokenized_sentence = tokenizer.texts_to_sequences([sentence])[0]

  for i in range(1,len(tokenized_sentence)):
    input_sequences.append(tokenized_sentence[:i+1])
```

[39]:

```
input_sequences
```

```
[39]: [[101, 14],
        [101, 14, 8],
        [101, 14, 8, 11],
        [101, 14, 8, 11, 4],
        [101, 14, 8, 11, 4, 53],
        [101, 14, 8, 11, 4, 53, 9],
        [101, 14, 8, 11, 4, 53, 9, 12],
        [101, 14, 8, 11, 4, 53, 9, 12, 54],
        [101, 14, 8, 11, 4, 53, 9, 12, 54, 102],
        [101, 14, 8, 11, 4, 53, 9, 12, 54, 102, 103],
        [101, 14, 8, 11, 4, 53, 9, 12, 54, 102, 103, 10],
        [8, 5],
        [8, 5, 105],
        [8, 5, 105, 106],
        [8, 5, 105, 106, 107],
        [8, 5, 105, 106, 107, 15],
        [8, 5, 105, 106, 107, 15, 33],
        [8, 5, 105, 106, 107, 15, 33, 34],
        [8, 5, 105, 106, 107, 15, 33, 34, 2],
        [8, 5, 105, 106, 107, 15, 33, 34, 2, 108],
        [8, 5, 105, 106, 107, 15, 33, 34, 2, 108, 13],
        [8, 5, 105, 106, 107, 15, 33, 34, 2, 108, 13, 21],
        [8, 5, 105, 106, 107, 15, 33, 34, 2, 108, 13, 21, 1],
        [8, 5, 105, 106, 107, 15, 33, 34, 2, 108, 13, 21, 1, 109],
        [8, 5, 105, 106, 107, 15, 33, 34, 2, 108, 13, 21, 1, 109, 4],
        [8, 5, 105, 106, 107, 15, 33, 34, 2, 108, 13, 21, 1, 109, 4, 55],
        [8, 5, 105, 106, 107, 15, 33, 34, 2, 108, 13, 21, 1, 109, 4, 55, 35],
        [8, 5, 105, 106, 107, 15, 33, 34, 2, 108, 13, 21, 1, 109, 4, 55, 35, 110],
        [8, 5, 105, 106, 107, 15, 33, 34, 2, 108, 13, 21, 1, 109, 4, 55, 35, 110, 56],
        [8,
         5,
         105,
         106,
         107,
         15,
         33,
         34,
         2,
         108,
         13,
         21,
         1,
         109,
         4,
         55,
         35,
         110,
  56,
  57],
 [8,
  5,
  105,
  106,
  107,
  15,
  33,
  34,
  2,
  108,
  13,
  21,
  1,
  109,
  4,
  55,
  35,
  110,
  56,
  57,
  111],
 [8,
  5,
  105,
  106,
  107,
  15,
  33,
  34,
  2,
  108,
  13,
  21,
  1,
  109,
  4,
  55,
  35,
  110,
  56,
  57,
  111,
  2],
 [8,
  5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3,
 115],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
  14,
  8,
  11,
  4,
  113,
  8,
  2,
  114,
  3,
  115,
  9],
 [8,
  5,
  105,
  106,
  107,
  15,
  33,
  34,
  2,
  108,
  13,
  21,
  1,
  109,
  4,
  55,
  35,
  110,
  56,
  57,
  111,
  2,
  112,
  14,
  8,
  11,
  4,
  113,
  8,
  2,
  114,
  3,
  115,
  9,
  12],
 [8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3,
 115,
 9,
 12,
 17],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3,
 115,
 9,
 12,
 17,
 54],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3,
 115,
 9,
 12,
 17,
 54,
 6],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3,
 115,
 9,
 12,
 17,
 54,
 6,
 116],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3,
 115,
 9,
 12,
 17,
 54,
 6,
 116,
 2],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3,
 115,
 9,
 12,
 17,
 54,
 6,
 116,
 2,
 58],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3,
 115,
 9,
 12,
 17,
 54,
 6,
 116,
 2,
 58,
 3],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3,
 115,
 9,
 12,
 17,
 54,
 6,
 116,
 2,
 58,
 3,
 22],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3,
 115,
 9,
 12,
 17,
 54,
 6,
 116,
 2,
 58,
 3,
 22,
 4],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3,
 115,
 9,
 12,
 17,
 54,
 6,
 116,
 2,
 58,
 3,
 22,
 4,
 36],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3,
 115,
 9,
 12,
 17,
 54,
 6,
 116,
 2,
 58,
 3,
 22,
 4,
 36,
 59],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3,
 115,
 9,
 12,
 17,
 54,
 6,
 116,
 2,
 58,
 3,
 22,
 4,
 36,
 59,
 23],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3,
 115,
 9,
 12,
 17,
 54,
 6,
 116,
 2,
 58,
 3,
 22,
 4,
 36,
 59,
 23,
 5],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
  57,
  111,
  2,
  112,
  14,
  8,
  11,
  4,
  113,
  8,
  2,
  114,
  3,
  115,
  9,
  12,
  17,
  54,
  6,
  116,
  2,
  58,
  3,
  22,
  4,
  36,
  59,
  23,
  5,
  117],
 [8,
  5,
  105,
  106,
  107,
  15,
  33,
  34,
  2,
  108,
  13,
  21,
  1,
  109,
  4,
  55,
  35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3,
 115,
 9,
 12,
 17,
 54,
 6,
 116,
 2,
 58,
 3,
 22,
 4,
 36,
 59,
 23,
 5,
 117,
 3],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3,
 115,
 9,
 12,
 17,
 54,
 6,
 116,
 2,
 58,
 3,
 22,
 4,
 36,
 59,
 23,
 5,
 117,
 3,
 60],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3,
 115,
 9,
 12,
 17,
 54,
 6,
 116,
 2,
 58,
 3,
 22,
 4,
 36,
 59,
 23,
 5,
 117,
 3,
 60,
 61],
[8,
 5,
 105,
 106,
 107,
15,
33,
34,
2,
108,
13,
21,
1,
109,
4,
55,
35,
110,
56,
57,
111,
2,
112,
14,
8,
11,
4,
113,
8,
2,
114,
3,
115,
9,
12,
17,
54,
6,
116,
2,
58,
3,
22,
4,
36,
59,
23,
5,
117,
3,
60,
61,
 118],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3,
 115,
 9,
 12,
 17,
 54,
 6,
 116,
 2,
 58,
 3,
 22,
 4,
 36,
 59,
 23,
 5,
 117,
 3,
 60,
 61,
 118,
 119],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3,
 115,
 9,
 12,
 17,
 54,
 6,
 116,
 2,
 58,
 3,
 22,
 4,
 36,
 59,
 23,
 5,
 117,
 3,
 60,
 61,
 118,
 119,
 24],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3,
 115,
 9,
 12,
 17,
 54,
 6,
 116,
 2,
 58,
 3,
 22,
 4,
 36,
 59,
 23,
 5,
 117,
 3,
 60,
 61,
 118,
 119,
 24,
 25],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3,
 115,
 9,
 12,
 17,
 54,
 6,
 116,
 2,
 58,
 3,
 22,
 4,
 36,
 59,
 23,
 5,
 117,
 3,
 60,
 61,
 118,
 119,
 24,
 25,
 120],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3,
 115,
 9,
 12,
 17,
 54,
 6,
 116,
 2,
 58,
 3,
 22,
 4,
 36,
 59,
 23,
 5,
 117,
 3,
 60,
 61,
 118,
 119,
 24,
 25,
 120,
 37],
[8,
5,
105,
106,
107,
15,
33,
34,
2,
108,
13,
21,
1,
109,
4,
55,
35,
110,
56,
57,
111,
2,
112,
14,
8,
11,
4,
113,
8,
2,
114,
3,
115,
9,
12,
17,
54,
6,
116,
2,
58,
3,
22,
4,
36,
59,
23,
5,
  117,
  3,
  60,
  61,
  118,
  119,
  24,
  25,
  120,
  37,
  10],
 [8,
  5,
  105,
  106,
  107,
  15,
  33,
  34,
  2,
  108,
  13,
  21,
  1,
  109,
  4,
  55,
  35,
  110,
  56,
  57,
  111,
  2,
  112,
  14,
  8,
  11,
  4,
  113,
  8,
  2,
  114,
  3,
  115,
  9,
  12,
  17,
 54,
 6,
 116,
 2,
 58,
 3,
 22,
 4,
 36,
 59,
 23,
 5,
 117,
 3,
 60,
 61,
 118,
 119,
 24,
 25,
 120,
 37,
 10,
 121],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3,
 115,
 9,
 12,
 17,
 54,
 6,
 116,
 2,
 58,
 3,
 22,
 4,
 36,
 59,
 23,
 5,
 117,
 3,
 60,
 61,
 118,
 119,
 24,
 25,
 120,
 37,
 10,
 121,
 17],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
108,
13,
21,
1,
109,
4,
55,
35,
110,
56,
57,
111,
2,
112,
14,
8,
11,
4,
113,
8,
2,
114,
3,
115,
9,
12,
17,
54,
6,
116,
2,
58,
3,
22,
4,
36,
59,
23,
5,
117,
3,
60,
61,
118,
119,
24,
25,
 120,
 37,
 10,
 121,
 17,
 18],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3,
 115,
 9,
 12,
 17,
 54,
 6,
 116,
 2,
 58,
 3,
 22,
 4,
 36,
 59,
 23,
 5,
 117,
 3,
 60,
 61,
 118,
 119,
 24,
 25,
 120,
 37,
 10,
 121,
 17,
 18,
 122],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3,
 115,
 9,
 12,
 17,
 54,
 6,
 116,
 2,
 58,
 3,
 22,
 4,
 36,
 59,
 23,
 5,
 117,
 3,
 60,
 61,
 118,
 119,
 24,
 25,
 120,
 37,
 10,
 121,
 17,
 18,
 122,
 62],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
2,
108,
13,
21,
1,
109,
4,
55,
35,
110,
56,
57,
111,
2,
112,
14,
8,
11,
4,
113,
8,
2,
114,
3,
115,
9,
12,
17,
54,
6,
116,
2,
58,
3,
22,
4,
36,
59,
23,
5,
117,
3,
60,
61,
118,
119,
24,
 25,
 120,
 37,
 10,
 121,
 17,
 18,
 122,
 62,
 2],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3,
 115,
 9,
 12,
 17,
 54,
  6,
  116,
  2,
  58,
  3,
  22,
  4,
  36,
  59,
  23,
  5,
  117,
  3,
  60,
  61,
  118,
  119,
  24,
  25,
  120,
  37,
  10,
  121,
  17,
  18,
  122,
  62,
  2,
  38],
 [8,
  5,
  105,
  106,
  107,
  15,
  33,
  34,
  2,
  108,
  13,
  21,
  1,
  109,
  4,
  55,
  35,
  110,
56,
57,
111,
2,
112,
14,
8,
11,
4,
113,
8,
2,
114,
3,
115,
9,
12,
17,
54,
6,
116,
2,
58,
3,
22,
4,
36,
59,
23,
5,
117,
3,
60,
61,
118,
119,
24,
25,
120,
37,
10,
121,
17,
18,
122,
62,
2,
 38,
 4],
[8,
 5,
 105,
 106,
 107,
 15,
 33,
 34,
 2,
 108,
 13,
 21,
 1,
 109,
 4,
 55,
 35,
 110,
 56,
 57,
 111,
 2,
 112,
 14,
 8,
 11,
 4,
 113,
 8,
 2,
 114,
 3,
 115,
 9,
 12,
 17,
 54,
 6,
 116,
 2,
 58,
 3,
 22,
 4,
 36,
 59,
 23,
 5,
 117,
 3,
 60,
 61,
 118,
 119,
 24,
 25,
 120,
 37,
 10,
 121,
 17,
 18,
 122,
 62,
 2,
 38,
 4,
 63],
[9, 4],
[9, 4, 35],
[9, 4, 35, 64],
[9, 4, 35, 64, 56],
[9, 4, 35, 64, 56, 124],
[9, 4, 35, 64, 56, 124, 26],
[9, 4, 35, 64, 56, 124, 26, 5],
[9, 4, 35, 64, 56, 124, 26, 5, 125],
[9, 4, 35, 64, 56, 124, 26, 5, 125, 65],
[9, 4, 35, 64, 56, 124, 26, 5, 125, 65, 6],
[9, 4, 35, 64, 56, 124, 26, 5, 125, 65, 6, 2],
[9, 4, 35, 64, 56, 124, 26, 5, 125, 65, 6, 2, 126],
[9, 4, 35, 64, 56, 124, 26, 5, 125, 65, 6, 2, 126, 4],
[9, 4, 35, 64, 56, 124, 26, 5, 125, 65, 6, 2, 126, 4, 66],
[9, 4, 35, 64, 56, 124, 26, 5, 125, 65, 6, 2, 126, 4, 66, 67],
[9, 4, 35, 64, 56, 124, 26, 5, 125, 65, 6, 2, 126, 4, 66, 67, 9],
[9, 4, 35, 64, 56, 124, 26, 5, 125, 65, 6, 2, 126, 4, 66, 67, 9, 39],
[9, 4, 35, 64, 56, 124, 26, 5, 125, 65, 6, 2, 126, 4, 66, 67, 9, 39, 127],
[9,
 4,
 35,
 64,
 56,
 124,
 26,
 5,
 125,
 65,
 6,
 2,
 126,
 4,
 66,
 67,
 9,
 39,
 127,
 128],
[9,
 4,
 35,
 64,
 56,
 124,
 26,
 5,
 125,
 65,
 6,
 2,
 126,
 4,
 66,
 67,
 9,
 39,
 127,
 128,
 40],
[9,
 4,
 35,
 64,
 56,
 124,
 26,
 5,
 125,
 65,
 6,
 2,
 126,
 4,
 66,
 67,
 9,
 39,
 127,
 128,
 40,
 129],
[9,
 4,
 35,
 64,
 56,
 124,
 26,
 5,
 125,
 65,
 6,
 2,
 126,
 4,
 66,
 67,
 9,
 39,
 127,
 128,
 40,
 129,
 8],
[9,
 4,
 35,
 64,
 56,
 124,
 26,
 5,
 125,
 65,
 6,
 2,
 126,
 4,
 66,
 67,
 9,
 39,
 127,
 128,
 40,
 129,
 8,
 130],
[9,
 4,
 35,
 64,
 56,
 124,
 26,
 5,
 125,
 65,
 6,
 2,
 126,
 4,
 66,
 67,
 9,
 39,
 127,
 128,
 40,
 129,
 8,
 130,
 68],
[9,
 4,
 35,
 64,
 56,
 124,
 26,
 5,
 125,
 65,
 6,
 2,
 126,
 4,
 66,
 67,
 9,
 39,
 127,
 128,
 40,
 129,
 8,
 130,
 68,
 131],
[9,
 4,
 35,
 64,
 56,
 124,
 26,
 5,
 125,
 65,
 6,
 2,
 126,
 4,
 66,
 67,
 9,
 39,
 127,
 128,
 40,
 129,
 8,
 130,
 68,
 131,
 132],
[9,
 4,
 35,
 64,
 56,
 124,
 26,
 5,
 125,
 65,
 6,
 2,
 126,
 4,
 66,
 67,
 9,
 39,
 127,
 128,
 40,
 129,
 8,
 130,
 68,
 131,
 132,
 10],
[9,
 4,
 35,
 64,
 56,
 124,
 26,
 5,
 125,
 65,
 6,
 2,
 126,
 4,
 66,
 67,
 9,
 39,
 127,
 128,
 40,
 129,
 8,
 130,
 68,
  131,
  132,
  10,
  1],
 [9,
  4,
  35,
  64,
  56,
  124,
  26,
  5,
  125,
  65,
  6,
  2,
  126,
  4,
  66,
  67,
  9,
  39,
  127,
  128,
  40,
  129,
  8,
  130,
  68,
  131,
  132,
  10,
  1,
  27],
 [9,
  4,
  35,
  64,
  56,
  124,
  26,
  5,
  125,
  65,
  6,
  2,
  126,
 4,
 66,
 67,
 9,
 39,
 127,
 128,
 40,
 129,
 8,
 130,
 68,
 131,
 132,
 10,
 1,
 27,
 133],
[9,
 4,
 35,
 64,
 56,
 124,
 26,
 5,
 125,
 65,
 6,
 2,
 126,
 4,
 66,
 67,
 9,
 39,
 127,
 128,
 40,
 129,
 8,
 130,
 68,
 131,
 132,
 10,
 1,
 27,
 133,
 134],
[9,
 4,
 35,
 64,
 56,
 124,
 26,
 5,
 125,
 65,
 6,
 2,
 126,
 4,
 66,
 67,
 9,
 39,
 127,
 128,
 40,
 129,
 8,
 130,
 68,
 131,
 132,
 10,
 1,
 27,
 133,
 134,
 19],
[9,
 4,
 35,
 64,
 56,
 124,
 26,
 5,
 125,
 65,
 6,
 2,
 126,
 4,
 66,
 67,
 9,
 39,
 127,
 128,
 40,
 129,
 8,
 130,
 68,
 131,
 132,
 10,
 1,
 27,
 133,
 134,
 19,
 135],
[9,
 4,
 35,
 64,
 56,
 124,
 26,
 5,
 125,
 65,
 6,
 2,
 126,
 4,
 66,
 67,
 9,
 39,
 127,
 128,
 40,
 129,
 8,
 130,
 68,
 131,
 132,
 10,
 1,
 27,
 133,
 134,
 19,
 135,
 2],
[9,
 4,
 35,
 64,
 56,
 124,
 26,
 5,
 125,
 65,
 6,
 2,
 126,
 4,
 66,
 67,
 9,
 39,
 127,
 128,
 40,
 129,
 8,
 130,
 68,
 131,
 132,
 10,
 1,
 27,
 133,
 134,
 19,
 135,
 2,
 136],
[9,
 4,
 35,
 64,
 56,
 124,
 26,
 5,
 125,
 65,
 6,
 2,
 126,
 4,
 66,
 67,
 9,
 39,
 127,
 128,
 40,
 129,
 8,
 130,
 68,
 131,
 132,
 10,
 1,
 27,
 133,
 134,
 19,
 135,
 2,
 136,
 3],
[9,
 4,
 35,
 64,
 56,
 124,
 26,
 5,
 125,
 65,
 6,
 2,
 126,
 4,
 66,
 67,
 9,
 39,
 127,
 128,
 40,
 129,
 8,
 130,
 68,
 131,
 132,
 10,
 1,
 27,
 133,
 134,
 19,
 135,
 2,
 136,
 3,
 137],
[9,
 4,
 35,
 64,
 56,
 124,
 26,
 5,
 125,
 65,
 6,
 2,
 126,
 4,
 66,
 67,
 9,
 39,
 127,
   128,
   40,
   129,
   8,
   130,
   68,
   131,
   132,
   10,
   1,
   27,
   133,
   134,
   19,
   135,
   2,
   136,
   3,
   137,
   138],
  [9,
   4,
   35,
   64,
   56,
   124,
   26,
   5,
   125,
   65,
   6,
   2,
   126,
   4,
   66,
   67,
   9,
   39,
   127,
   128,
   40,
   129,
```

[40]:
```python
max_len = max([len(x) for x in input_sequences])
max_len
```

[40]: 83

```
   131,
   132,
   10,
   1,
   27,
   133,
   134,
   19,
[41]: from tensorflow.keras.preprocessing.sequence import pad_sequences
      padded_input_sequences = pad_sequences(input_sequences, maxlen = max_len,␣
        ↪padding='pre')
```

The code you provided is using the pad_sequences function from the tensorflow.keras.preprocessing.sequence module. This function is typically used to pad sequences to ensure uniform length, which is often necessary when working with neural networks.

Here's what each part of the code does:

from tensorflow.keras.preprocessing.sequence import pad_sequences: This line imports the pad_sequences function from the tensorflow.keras.preprocessing.sequence module. This function is used to pad sequences to a maximum length.

padded_input_sequences = pad_sequences(input_sequences, maxlen=max_len, padding='pre'): This line applies padding to the input_sequences array.

input_sequences: This is assumed to be a list of sequences (lists of integers).

maxlen: This parameter specifies the maximum length of sequences after padding. Sequences longer than this length will be truncated, and sequences shorter than this length will be padded.

padding: This parameter specifies whether to pad sequences at the beginning ('pre') or at the end ('post'). In this case, it pads sequences at the beginning.

The result, padded_input_sequences, will be a 2D numpy array where each row corresponds to a sequence, and the sequences are padded or truncated to have a length of max_len.

```
[42]: padded_input_sequences
```

```
[42]: array([[  0,   0,   0, …,   0, 101,  14],
             [  0,   0,   0, …, 101,  14,   8],
             [  0,   0,   0, …,  14,   8,  11],
             …,
             [  0,   0,   2, …,  21, 343,  13],
             [  0,   2,  31, …, 343,  13,  28],
             [  2,  31,   3, …,  13,  28, 100]], dtype=int32)
```

```
[43]: X = padded_input_sequences[:,:-1]

      X
```

```
[43]: array([[  0,   0,   0, …,   0,   0, 101],
             [  0,   0,   0, …,   0, 101,  14],
             [  0,   0,   0, …, 101,  14,   8],
             …,
             [  0,   0,   2, …,   5,  21, 343],
             [  0,   2,  31, …,  21, 343,  13],
             [  2,  31,   3, …, 343,  13,  28]], dtype=int32)
```

```
[44]: y = padded_input_sequences[:,-1]
```

[45]:
```
X.shape
```

[45]:
```
(682, 82)
```

[46]:
```
y.shape
```

[46]:
```
(682,)
```

[47]:
```
from tensorflow.keras.utils import to_categorical
y = to_categorical(y,num_classes=344)
```

[48]:
```
y
```

[48]:
```
array([[0., 0., 0., …, 0., 0., 0.],
       [0., 0., 0., …, 0., 0., 0.],
       [0., 0., 0., …, 0., 0., 0.],
       …,
       [0., 0., 0., …, 0., 0., 0.],
       [0., 0., 0., …, 0., 0., 0.],
       [0., 0., 0., …, 0., 0., 0.]], dtype=float32)
```

[49]:
```
y.shape
```

[49]:
```
(682, 344)
```

[50]:
```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Embedding, LSTM, Dense
```

[51]:
```
model = Sequential()
model.add(Embedding(344, 200, input_length=82))
model.add(LSTM(150, return_sequences=True, input_shape=(82, 200)))
model.add(LSTM(150))
model.add(Dense(344, activation='softmax'))
```

[52]:
```
model.compile(loss='categorical_crossentropy',␣
  ↪optimizer='adam',metrics=['accuracy'])
```

[53]:
```
model.fit(X,y,epochs=100)
```

```
Epoch 1/100
22/22 [==============================] - 11s 153ms/step - loss: 5.7070 -
accuracy: 0.0220
Epoch 2/100
22/22 [==============================] - 3s 113ms/step - loss: 5.3990 -
accuracy: 0.0308
Epoch 3/100
22/22 [==============================] - 2s 87ms/step - loss: 5.3145 - accuracy:
```

```
0.0293
Epoch 4/100
22/22 [==============================] - 1s 68ms/step - loss: 5.2843 - accuracy:
0.0469
Epoch 5/100
22/22 [==============================] - 1s 63ms/step - loss: 5.2770 - accuracy:
0.0469
Epoch 6/100
22/22 [==============================] - 2s 107ms/step - loss: 5.2637 -
accuracy: 0.0455
Epoch 7/100
22/22 [==============================] - 1s 42ms/step - loss: 5.2524 - accuracy:
0.0440
Epoch 8/100
22/22 [==============================] - 1s 56ms/step - loss: 5.2121 - accuracy:
0.0484
Epoch 9/100
22/22 [==============================] - 1s 44ms/step - loss: 5.1350 - accuracy:
0.0557
Epoch 10/100
22/22 [==============================] - 1s 27ms/step - loss: 5.0367 - accuracy:
0.0572
Epoch 11/100
22/22 [==============================] - 0s 11ms/step - loss: 4.9326 - accuracy:
0.0587
Epoch 12/100
22/22 [==============================] - 1s 24ms/step - loss: 4.8010 - accuracy:
0.0689
Epoch 13/100
22/22 [==============================] - 1s 23ms/step - loss: 4.6624 - accuracy:
0.0792
Epoch 14/100
22/22 [==============================] - 1s 31ms/step - loss: 4.5219 - accuracy:
0.0718
Epoch 15/100
22/22 [==============================] - 1s 24ms/step - loss: 4.3471 - accuracy:
0.0953
Epoch 16/100
22/22 [==============================] - 1s 18ms/step - loss: 4.2101 - accuracy:
0.1070
Epoch 17/100
22/22 [==============================] - 1s 26ms/step - loss: 4.0468 - accuracy:
0.0968
Epoch 18/100
22/22 [==============================] - 1s 24ms/step - loss: 3.9155 - accuracy:
0.1246
Epoch 19/100
22/22 [==============================] - 0s 17ms/step - loss: 3.7425 - accuracy:
```

```
0.1452
Epoch 20/100
22/22 [==============================] - 0s 17ms/step - loss: 3.5970 - accuracy:
0.1730
Epoch 21/100
22/22 [==============================] - 1s 24ms/step - loss: 3.4499 - accuracy:
0.1921
Epoch 22/100
22/22 [==============================] - 1s 24ms/step - loss: 3.3357 - accuracy:
0.2287
Epoch 23/100
22/22 [==============================] - 1s 24ms/step - loss: 3.1837 - accuracy:
0.2639
Epoch 24/100
22/22 [==============================] - 1s 35ms/step - loss: 3.0569 - accuracy:
0.3050
Epoch 25/100
22/22 [==============================] - 0s 13ms/step - loss: 2.9333 - accuracy:
0.3548
Epoch 26/100
22/22 [==============================] - 1s 24ms/step - loss: 2.8015 - accuracy:
0.3856
Epoch 27/100
22/22 [==============================] - 0s 23ms/step - loss: 2.6950 - accuracy:
0.4194
Epoch 28/100
22/22 [==============================] - 0s 17ms/step - loss: 2.5602 - accuracy:
0.4633
Epoch 29/100
22/22 [==============================] - 1s 24ms/step - loss: 2.4488 - accuracy:
0.4971
Epoch 30/100
22/22 [==============================] - 1s 25ms/step - loss: 2.3403 - accuracy:
0.5147
Epoch 31/100
22/22 [==============================] - 0s 17ms/step - loss: 2.2290 - accuracy:
0.5762
Epoch 32/100
22/22 [==============================] - 0s 10ms/step - loss: 2.1150 - accuracy:
0.6012
Epoch 33/100
22/22 [==============================] - 0s 18ms/step - loss: 2.0162 - accuracy:
0.6437
Epoch 34/100
22/22 [==============================] - 0s 17ms/step - loss: 1.9216 - accuracy:
0.6554
Epoch 35/100
22/22 [==============================] - 1s 24ms/step - loss: 1.8282 - accuracy:
```

```
0.6965
Epoch 36/100
22/22 [==============================] - 0s 10ms/step - loss: 1.7400 - accuracy:
0.7170
Epoch 37/100
22/22 [==============================] - 0s 10ms/step - loss: 1.6506 - accuracy:
0.7258
Epoch 38/100
22/22 [==============================] - 0s 17ms/step - loss: 1.5698 - accuracy:
0.7581
Epoch 39/100
22/22 [==============================] - 0s 17ms/step - loss: 1.4959 - accuracy:
0.7713
Epoch 40/100
22/22 [==============================] - 0s 17ms/step - loss: 1.4166 - accuracy:
0.8079
Epoch 41/100
22/22 [==============================] - 0s 17ms/step - loss: 1.3454 - accuracy:
0.8167
Epoch 42/100
22/22 [==============================] - 0s 17ms/step - loss: 1.2734 - accuracy:
0.8226
Epoch 43/100
22/22 [==============================] - 1s 24ms/step - loss: 1.2104 - accuracy:
0.8475
Epoch 44/100
22/22 [==============================] - 0s 10ms/step - loss: 1.1521 - accuracy:
0.8475
Epoch 45/100
22/22 [==============================] - 1s 24ms/step - loss: 1.0918 - accuracy:
0.8695
Epoch 46/100
22/22 [==============================] - 0s 10ms/step - loss: 1.0417 - accuracy:
0.8783
Epoch 47/100
22/22 [==============================] - 0s 10ms/step - loss: 0.9954 - accuracy:
0.8812
Epoch 48/100
22/22 [==============================] - 0s 17ms/step - loss: 0.9453 - accuracy:
0.8915
Epoch 49/100
22/22 [==============================] - 0s 17ms/step - loss: 0.8952 - accuracy:
0.8988
Epoch 50/100
22/22 [==============================] - 0s 10ms/step - loss: 0.8491 - accuracy:
0.9076
Epoch 51/100
22/22 [==============================] - 0s 10ms/step - loss: 0.8087 - accuracy:
```

```
0.9164
Epoch 52/100
22/22 [==============================] - 0s 17ms/step - loss: 0.7701 - accuracy:
0.9194
Epoch 53/100
22/22 [==============================] - 0s 10ms/step - loss: 0.7325 - accuracy:
0.9296
Epoch 54/100
22/22 [==============================] - 0s 17ms/step - loss: 0.6946 - accuracy:
0.9296
Epoch 55/100
22/22 [==============================] - 0s 10ms/step - loss: 0.6648 - accuracy:
0.9370
Epoch 56/100
22/22 [==============================] - 0s 13ms/step - loss: 0.6325 - accuracy:
0.9340
Epoch 57/100
22/22 [==============================] - 0s 13ms/step - loss: 0.6046 - accuracy:
0.9443
Epoch 58/100
22/22 [==============================] - 0s 13ms/step - loss: 0.5753 - accuracy:
0.9516
Epoch 59/100
22/22 [==============================] - 0s 13ms/step - loss: 0.5422 - accuracy:
0.9560
Epoch 60/100
22/22 [==============================] - 0s 13ms/step - loss: 0.5227 - accuracy:
0.9487
Epoch 61/100
22/22 [==============================] - 0s 13ms/step - loss: 0.5091 - accuracy:
0.9501
Epoch 62/100
22/22 [==============================] - 1s 37ms/step - loss: 0.4756 - accuracy:
0.9619
Epoch 63/100
22/22 [==============================] - 1s 26ms/step - loss: 0.4512 - accuracy:
0.9589
Epoch 64/100
22/22 [==============================] - 0s 10ms/step - loss: 0.4317 - accuracy:
0.9663
Epoch 65/100
22/22 [==============================] - 0s 17ms/step - loss: 0.4113 - accuracy:
0.9663
Epoch 66/100
22/22 [==============================] - 0s 17ms/step - loss: 0.3972 - accuracy:
0.9663
Epoch 67/100
22/22 [==============================] - 0s 17ms/step - loss: 0.3795 - accuracy:
```

```
0.9677
Epoch 68/100
22/22 [==============================] - 0s 18ms/step - loss: 0.3622 - accuracy:
0.9677
Epoch 69/100
22/22 [==============================] - 0s 18ms/step - loss: 0.3452 - accuracy:
0.9633
Epoch 70/100
22/22 [==============================] - 0s 17ms/step - loss: 0.3317 - accuracy:
0.9721
Epoch 71/100
22/22 [==============================] - 0s 10ms/step - loss: 0.3212 - accuracy:
0.9692
Epoch 72/100
22/22 [==============================] - 0s 10ms/step - loss: 0.3072 - accuracy:
0.9707
Epoch 73/100
22/22 [==============================] - 0s 17ms/step - loss: 0.2930 - accuracy:
0.9736
Epoch 74/100
22/22 [==============================] - 0s 17ms/step - loss: 0.2824 - accuracy:
0.9736
Epoch 75/100
22/22 [==============================] - 0s 10ms/step - loss: 0.2735 - accuracy:
0.9721
Epoch 76/100
22/22 [==============================] - 0s 17ms/step - loss: 0.2631 - accuracy:
0.9736
Epoch 77/100
22/22 [==============================] - 0s 17ms/step - loss: 0.2555 - accuracy:
0.9707
Epoch 78/100
22/22 [==============================] - 0s 10ms/step - loss: 0.2456 - accuracy:
0.9721
Epoch 79/100
22/22 [==============================] - 0s 9ms/step - loss: 0.2344 - accuracy:
0.9721
Epoch 80/100
22/22 [==============================] - 0s 10ms/step - loss: 0.2285 - accuracy:
0.9765
Epoch 81/100
22/22 [==============================] - 0s 17ms/step - loss: 0.2217 - accuracy:
0.9736
Epoch 82/100
22/22 [==============================] - 1s 32ms/step - loss: 0.2142 - accuracy:
0.9751
Epoch 83/100
22/22 [==============================] - 0s 17ms/step - loss: 0.2069 - accuracy:
```

```
0.9809
Epoch 84/100
22/22 [==============================] - 0s 10ms/step - loss: 0.2018 - accuracy:
0.9721
Epoch 85/100
22/22 [==============================] - 0s 10ms/step - loss: 0.1943 - accuracy:
0.9809
Epoch 86/100
22/22 [==============================] - 0s 17ms/step - loss: 0.1864 - accuracy:
0.9780
Epoch 87/100
22/22 [==============================] - 0s 10ms/step - loss: 0.1806 - accuracy:
0.9809
Epoch 88/100
22/22 [==============================] - 0s 10ms/step - loss: 0.1766 - accuracy:
0.9780
Epoch 89/100
22/22 [==============================] - 0s 17ms/step - loss: 0.1711 - accuracy:
0.9795
Epoch 90/100
22/22 [==============================] - 0s 10ms/step - loss: 0.1668 - accuracy:
0.9795
Epoch 91/100
22/22 [==============================] - 0s 17ms/step - loss: 0.1608 - accuracy:
0.9765
Epoch 92/100
22/22 [==============================] - 0s 10ms/step - loss: 0.1579 - accuracy:
0.9795
Epoch 93/100
22/22 [==============================] - 0s 10ms/step - loss: 0.1526 - accuracy:
0.9780
Epoch 94/100
22/22 [==============================] - 0s 17ms/step - loss: 0.1495 - accuracy:
0.9809
Epoch 95/100
22/22 [==============================] - 0s 11ms/step - loss: 0.1458 - accuracy:
0.9839
Epoch 96/100
22/22 [==============================] - 0s 14ms/step - loss: 0.1410 - accuracy:
0.9824
Epoch 97/100
22/22 [==============================] - 1s 23ms/step - loss: 0.1372 - accuracy:
0.9809
Epoch 98/100
22/22 [==============================] - 0s 14ms/step - loss: 0.1340 - accuracy:
0.9809
Epoch 99/100
22/22 [==============================] - 1s 24ms/step - loss: 0.1320 - accuracy:
```

```
0.9839
Epoch 100/100
22/22 [==============================] - 1s 23ms/step - loss: 0.1277 - accuracy:
0.9839
```

```
[53]: <keras.src.callbacks.History at 0x7a1a543d6fb0>
```

```
[54]: import time
import numpy as np
text = "At the heart of SolarMax"

for i in range(10):
  # tokenize
  token_text = tokenizer.texts_to_sequences([text])[0]
  # padding
  padded_token_text = pad_sequences([token_text], maxlen=82, padding='pre')
  # predict
  pos = np.argmax(model.predict(padded_token_text))

  for word,index in tokenizer.word_index.items():
    if index == pos:
      text = text + " " + word
      print(text)
      time.sleep(2)
```

```
1/1 [==============================] - 2s 2s/step
At the heart of SolarMax lies
1/1 [==============================] - 0s 59ms/step
At the heart of SolarMax lies a
1/1 [==============================] - 0s 102ms/step
At the heart of SolarMax lies a sophisticated
1/1 [==============================] - 0s 19ms/step
At the heart of SolarMax lies a sophisticated nanomaterial
1/1 [==============================] - 0s 19ms/step
At the heart of SolarMax lies a sophisticated nanomaterial that
1/1 [==============================] - 0s 19ms/step
At the heart of SolarMax lies a sophisticated nanomaterial that exhibits
1/1 [==============================] - 0s 27ms/step
At the heart of SolarMax lies a sophisticated nanomaterial that exhibits
exceptional
1/1 [==============================] - 0s 20ms/step
At the heart of SolarMax lies a sophisticated nanomaterial that exhibits
exceptional light
1/1 [==============================] - 0s 19ms/step
At the heart of SolarMax lies a sophisticated nanomaterial that exhibits
exceptional light absorbing
1/1 [==============================] - 0s 19ms/step
At the heart of SolarMax lies a sophisticated nanomaterial that exhibits
```

```
exceptional light absorbing properties
```