Experiment 7: Summarize long texts using a pre-trained summarization model using Hugging face model.

- Load the summarization pipeline.
- Take a passage as input and obtain the summarized text.

AIM: To summarize long text using a pre-trained Hugging Face model by loading the summarization pipeline, providing a passage as input, and generating a concise output.

Introduction:

Text summarization is a key task in Natural Language Processing (NLP) that condenses a large block of text into a short, meaningful summary. It is especially useful for applications like news summarization, document previews, and information retrieval. In this lab, we explore summarization using an **abstractive model**, which generates new phrases and sentences rather than merely extracting existing ones.

The model used is facebook/bart-large-cnn, an advanced transformer-based sequence-to-sequence model developed by Facebook AI. It excels at summarization by encoding the input text and decoding it into a shorter, human-like version.

We use the transformers library provided by **Hugging Face**, a powerful open-source platform that simplifies the use of state-of-the-art machine learning models. It offers a high-level API called pipeline() that abstracts the complexity of loading pre-trained models, managing tokenization, and performing inference, all in a few lines of code. A public online repository of pre-trained model where we can browse, search, and test models.can be found in URL: https://huggingface.co/models

Using Hugging Face offers several benefits:

Access to Pre-trained Models:
Hugging Face provides thousands of models that are already trained on large datasets. This saves time and resources—you don't have to train models from scratch.

Easy-to-use Pipeline API:
With just one line of code (e.g., pipeline("summarization")), you can load a full NLP system including the model and tokenizer. This simplifies development and reduces coding effort.

High Accuracy with Less Effort:
The models are trained by top research labs (like Facebook AI, Google, etc.), meaning they are well-optimized and deliver strong performance out-of-the-box.

Supports Multiple Languages and Tasks:
Whether you're summarizing English text, analyzing Hindi tweets, or translating French paragraphs—Hugging Face has models for it.

Integration with Google Colab and Google Drive:
You can easily run these models in the cloud (Colab), and even store them in Google Drive for reusability.

Community and Documentation:
Hugging Face has a large community of contributors and excellent documentation, making it easier for beginners and researchers to learn and experiment.

Key Concepts Introduced:

1. **Text Summarization:** The process of shortening a long document into its essential meaning.
2. **Abstractive Summarization:** Generating new sentences rather than extracting directly.
3. **Hugging Face Transformers:** An open-source Python library with access to thousands of pre-trained NLP models.
4. **BART:** Facebook's encoder-decoder transformer for text generation and summarization.
5. **Google Colab + Google Drive:** Cloud-based coding with persistent model storage.

Objectives of the Program

1. To understand abstractive summarization using a pre-trained model.
2. To use the Hugging Face pipeline for summarization.
3. To read long text content from a file.
4. To generate and display the summarized version of the input.
5. To store the model in Google Drive and avoid repeated downloads.

Step 1: Install Required Libraries

```
!pip install transformers sentencepiece --quiet
```

Step 2: Import libraries

```
from transformers import pipeline
from google.colab import files, drive
import os
```

Step 3: Mount Google Drive

```
drive.mount('/content/drive')
```

Output: Mounted at /content/drive

Step 4: Define model save path inside Google Drive

```
model_dir = "/content/drive/MyDrive/bart_summarizer"
```

Step 5: Load or download the model

```
if os.path.exists(model_dir):
    print("Loading model from Google Drive...")
    summarizer = pipeline("summarization", model=model_dir, tokenizer=model_dir)
else:
    print("Downloading model from Hugging Face for the first time...")
```

```
summarizer = pipeline("summarization", model="facebook/bart-large-cnn")

summarizer.model.save_pretrained(model_dir)

summarizer.tokenizer.save_pretrained(model_dir)

print("Model downloaded and saved to Google Drive")
```

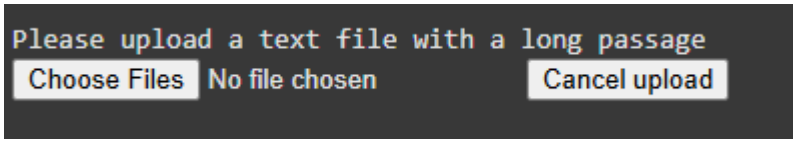
Output: Downloading model from Hugging Face for the first time...

Model downloaded and saved to Google Drive

Step 6: Upload a text file

```
print("\nPlease upload a text file with a long passage")
uploaded_file = files.upload()
```

Output



Step 7: Read the uploaded file

```
file_name = list(uploaded_file.keys())[0]
with open(file_name, 'r') as file:
    input_text = file.read()
```

Step 8: Summarize the text file

```
print("\n Summarizing... please wait.")
summary = summarizer(input_text, max_length=150, min_length=40, truncation=True)
summary_text = summary[0]['summary_text']
print("Summarization completed successfully")
```

Output: Summarizing... please wait.

Summarization completed successfully

Step 9: Display output

```
print("\n--- Original Text (First 500 characters) ---")
print(input_text[:500] + "..." if len(input_text) > 500 else input_text)
print("\n--- Summarized Text ---")
print(summary_text)
```

--- Original Text (First 500 characters) ---

Artificial Intelligence (AI) is rapidly transforming the landscape of modern education, introducing a paradigm shift in how knowledge is delivered, absorbed, and assessed. One of the most significant contributions of AI is the advent of adaptive learning techniques. These systems analyze vast amounts of student data, including individual learning paces, existing strengths, and areas of weakness, to create truly personalized educational experiences. This level of customization allows students to ...

--- Summarized Text ---

Artificial Intelligence (AI) is rapidly transforming the landscape of modern education. AI-powered chatbots and virtual assistants are becoming increasingly common within academic institutions. AI plays a crucial role in enhancing accessibility and inclusivity in education.

Final Note:

In this experiment, we use the pre-trained model facebook/bart-large-cnn for text summarization. The model is based on the **BART transformer architecture**, developed and trained by **Facebook AI**. This trained model is hosted publicly on the **Hugging Face Model Hub**, can be found in URL: https://huggingface.co/models

The program uses the **Hugging Face transformers library** to access and run the model with the following components.

- **Tokenizer and tokenization logic**: Provided by Hugging Face.
  It converts input text into numerical token IDs that the model can understand (encoding), and later converts the model's output tokens back into readable text (decoding).
- **Model architecture and weights**: Developed and pre-trained by Facebook AI.
  These weights represent the learning the model has acquired and are specific to the summarization task. They are downloaded from Hugging Face's model hub using the model name facebook/bart-large-cnn.
- **Pipeline function**: Provided by Hugging Face's transformers library.
  It bundles the tokenizer and model into a single easy-to-use interface, abstracting away complex setup steps.

The **tokenizer and code logic** come from Hugging Face, while the **core model and trained weights** come from Facebook AI, all integrated seamlessly using the transformers library and accessed via the Hugging Face platform.