

✓ AI & Generative AI Internship Project

Name: Sundaram

Project Title: Text Generation using a Pre-trained AI Model

Tools Used: Google Colab, Python, Hugging Face Transformers

Description:

This project demonstrates the use of Generative AI for text generation.

We apply a pre-trained language model to automatically generate human-like text from given prompts.

```
!pip install transformers torch
```

```
Requirement already satisfied: transformers in /usr/local/lib/python3.12/dist-packages (4.55.2)
Requirement already satisfied: torch in /usr/local/lib/python3.12/dist-packages (2.8.0+cu126)
Requirement already satisfied: filelock in /usr/local/lib/python3.12/dist-packages (from transformers) (3.19.1)
Requirement already satisfied: huggingface-hub<1.0,>=0.34.0 in /usr/local/lib/python3.12/dist-packages (from transformers) (0.25.0)
Requirement already satisfied: numpy>=1.17 in /usr/local/lib/python3.12/dist-packages (from transformers) (2.0.2)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.12/dist-packages (from transformers) (25.0)
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.12/dist-packages (from transformers) (6.0.2)
Requirement already satisfied: regex!=2019.12.17 in /usr/local/lib/python3.12/dist-packages (from transformers) (2024.11.6)
Requirement already satisfied: requests in /usr/local/lib/python3.12/dist-packages (from transformers) (2.32.4)
Requirement already satisfied: tokenizers<0.22,>=0.21 in /usr/local/lib/python3.12/dist-packages (from transformers) (0.21.4)
Requirement already satisfied: safetensors>=0.4.3 in /usr/local/lib/python3.12/dist-packages (from transformers) (0.6.2)
Requirement already satisfied: tqdm>=4.27 in /usr/local/lib/python3.12/dist-packages (from transformers) (4.67.1)
Requirement already satisfied: typing-extensions>=4.10.0 in /usr/local/lib/python3.12/dist-packages (from torch) (4.14.1)
Requirement already satisfied: setuptools in /usr/local/lib/python3.12/dist-packages (from torch) (75.2.0)
Requirement already satisfied: sympy>=1.13.3 in /usr/local/lib/python3.12/dist-packages (from torch) (1.13.3)
Requirement already satisfied: networkx in /usr/local/lib/python3.12/dist-packages (from torch) (3.5)
Requirement already satisfied: Jinja2 in /usr/local/lib/python3.12/dist-packages (from torch) (3.1.6)
Requirement already satisfied: fsspec in /usr/local/lib/python3.12/dist-packages (from torch) (2025.3.0)
Requirement already satisfied: nvidia-cuda-nvrtc-cu12==12.6.77 in /usr/local/lib/python3.12/dist-packages (from torch) (12.6.77)
Requirement already satisfied: nvidia-cuda-runtime-cu12==12.6.77 in /usr/local/lib/python3.12/dist-packages (from torch) (12.6.77)
Requirement already satisfied: nvidia-cuda-cupti-cu12==12.6.80 in /usr/local/lib/python3.12/dist-packages (from torch) (12.6.80)
Requirement already satisfied: nvidia-cudnn-cu12==9.10.2.21 in /usr/local/lib/python3.12/dist-packages (from torch) (9.10.2.21)
Requirement already satisfied: nvidia-cublas-cu12==12.6.4.1 in /usr/local/lib/python3.12/dist-packages (from torch) (12.6.4.1)
Requirement already satisfied: nvidia-cufft-cu12==11.3.0.4 in /usr/local/lib/python3.12/dist-packages (from torch) (11.3.0.4)
Requirement already satisfied: nvidia-curand-cu12==10.3.7.77 in /usr/local/lib/python3.12/dist-packages (from torch) (10.3.7.77)
Requirement already satisfied: nvidia-cusolver-cu12==11.7.1.2 in /usr/local/lib/python3.12/dist-packages (from torch) (11.7.1.2)
Requirement already satisfied: nvidia-cusparselt-cu12==0.7.1 in /usr/local/lib/python3.12/dist-packages (from torch) (0.7.1)
Requirement already satisfied: nvidia-nccl-cu12==2.27.3 in /usr/local/lib/python3.12/dist-packages (from torch) (2.27.3)
Requirement already satisfied: nvidia-nvtx-cu12==12.6.77 in /usr/local/lib/python3.12/dist-packages (from torch) (12.6.77)
Requirement already satisfied: nvidia-nvjitlink-cu12==12.6.85 in /usr/local/lib/python3.12/dist-packages (from torch) (12.6.85)
Requirement already satisfied: nvidia-cufile-cu12==1.11.1.6 in /usr/local/lib/python3.12/dist-packages (from torch) (1.11.1.6)
Requirement already satisfied: triton==3.4.0 in /usr/local/lib/python3.12/dist-packages (from torch) (3.4.0)
Requirement already satisfied: hf-xet<2.0.0,>=1.1.3 in /usr/local/lib/python3.12/dist-packages (from huggingface-hub<1.0,>=0.34.0->transformers) (1.1.3)
Requirement already satisfied: mpmath<1.4,>=1.1.0 in /usr/local/lib/python3.12/dist-packages (from sympy>=1.13.3->torch) (1.3.6)
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.12/dist-packages (from Jinja2->torch) (3.0.2)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.12/dist-packages (from requests->transformers) (3.4.0)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.12/dist-packages (from requests->transformers) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.12/dist-packages (from requests->transformers) (2.31.0)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.12/dist-packages (from requests->transformers) (2025.11.11)
```

```
from transformers import pipeline
```

```
text_generator = pipeline("text-generation", model="gpt2")
```

```

/usr/local/lib/python3.12/dist-packages/huggingface_hub/utils/_auth.py:94: UserWarning:
The secret `HF_TOKEN` does not exist in your Colab secrets.
To authenticate with the Hugging Face Hub, create a token in your settings tab (https://huggingface.co/settings/tokens), set it as an environment variable `HF_TOKEN`, or store it as a Colab secret.
You will be able to reuse this secret in all of your notebooks.
Please note that authentication is recommended but still optional to access public models or datasets.
warnings.warn(

config.json: 100% 665/665 [00:00<00:00, 14.1kB/s]

model.safetensors: 100% 548M/548M [00:08<00:00, 124MB/s]

generation_config.json: 100% 124/124 [00:00<00:00, 6.56kB/s]

tokenizer_config.json: 100% 26.0/26.0 [00:00<00:00, 1.43kB/s]

vocab.json: 100% 1.04M/1.04M [00:00<00:00, 11.3MB/s]

merges.txt: 100% 456k/456k [00:00<00:00, 9.42MB/s]

tokenizer.json: 100% 1.36M/1.36M [00:00<00:00, 14.2MB/s]

Device set to use cpu

```

```
prompt = "Artificial Intelligence is transforming industries because"  
output = text_generator(prompt, max_length=60, num_return_sequences=1)  
  
print(output[0]['generated_text'])
```

Truncation was not explicitly activated but `max_length` is provided a specific value, please use `truncation=True` to explicitly set the maximum length of the returned text. Setting `pad_token_id` to `eos_token_id`:50256 for open-end generation. Both `max_new_tokens` (=256) and `max_length` (=60) seem to have been set. `max_new_tokens` will take precedence. Please refer to the documentation for more details. Artificial Intelligence is transforming industries because of the data it gathers."

The research, published in a study by the University of Toronto, is the latest development in artificial intelligence research. Since 2005, AI companies have been working on developing AI-enabled products that could enable them to handle more jobs and improve efficiency. The research shows that as well as being able to identify and track the movements of humans in real-time, the technology can also be used to predict future movements. "This is not a new field, but it does seem to be a great opportunity for AI companies to offer more flexible and efficient solutions." "This is a big start, and we are looking forward to learning more about the applications of AI in future." The research comes to light after the discovery of a "network of sensors" that can identify and track all kinds of objects and people. In a recent

Evaluation

The model is able to generate coherent text from a short prompt. It produces meaningful sentences, although sometimes the results may drift off-topic. This shows both the usefulness and current limitations of generative AI models.

Conclusion

This project demonstrates how generative AI can be used for text generation. Through this work, I learned:

- How to use a pre-trained AI model for natural language tasks.
- How to run experiments in Google Colab.
- How to document and present AI projects effectively.

In the future, such models could be fine-tuned on specific datasets to improve accuracy and relevance for particular industries.