

```
!pip install transformers torch streamlit matplotlib
```

```

211.5/211.5 MB 5.6 MB/s eta 0:00:00
Downloading nvidia_curand_cu12-10.3.5.147-py3-none-manylinux2014_x86_64.whl (56.3 MB)
56.3/56.3 MB 14.6 MB/s eta 0:00:00
Downloading nvidia_cusolver_cu12-11.6.1.9-py3-none-manylinux2014_x86_64.whl (127.9 MB)
127.9/127.9 MB 7.3 MB/s eta 0:00:00
Downloading nvidia_cusparses_cu12-12.3.1.170-py3-none-manylinux2014_x86_64.whl (207.5 MB)
207.5/207.5 MB 6.6 MB/s eta 0:00:00
Downloading nvidia_nvjitlink_cu12-12.4.127-py3-none-manylinux2014_x86_64.whl (21.1 MB)
21.1/21.1 MB 94.4 MB/s eta 0:00:00
Downloading streamlit-1.45.1-py3-none-any.whl (9.9 MB)
9.9/9.9 MB 92.6 MB/s eta 0:00:00
Downloading pydeck-0.9.1-py2.py3-none-any.whl (6.9 MB)
6.9/6.9 MB 112.3 MB/s eta 0:00:00
Downloading watchdog-6.0.0-py3-none-manylinux2014_x86_64.whl (79 kB)
79.1/79.1 kB 6.1 MB/s eta 0:00:00
Installing collected packages: watchdog, nvidia-nvjitlink-cu12, nvidia-curand-cu12, nvidia-cufft-cu12, nvidia-cuda-runtime-cu12, nvidia-cuda-nvrtc-cu12, nvidia-cuda-cupti-cu12, nvidia-cublas-cu12, nvidia-cusparses-cu12, nvidia-cudnn-cu12, nvidia-cusolver-cu12
Attempting uninstall: nvidia-nvjitlink-cu12
Found existing installation: nvidia-nvjitlink-cu12 12.5.82
Uninstalling nvidia-nvjitlink-cu12-12.5.82:
Successfully uninstalled nvidia-nvjitlink-cu12-12.5.82
Attempting uninstall: nvidia-curand-cu12
Found existing installation: nvidia-curand-cu12 10.3.6.82
Uninstalling nvidia-curand-cu12-10.3.6.82:
Successfully uninstalled nvidia-curand-cu12-10.3.6.82
Attempting uninstall: nvidia-cufft-cu12
Found existing installation: nvidia-cufft-cu12 11.2.3.61
Uninstalling nvidia-cufft-cu12-11.2.3.61:
Successfully uninstalled nvidia-cufft-cu12-11.2.3.61
Attempting uninstall: nvidia-cuda-runtime-cu12
Found existing installation: nvidia-cuda-runtime-cu12 12.5.82
Uninstalling nvidia-cuda-runtime-cu12-12.5.82:
Successfully uninstalled nvidia-cuda-runtime-cu12-12.5.82
Attempting uninstall: nvidia-cuda-nvrtc-cu12
Found existing installation: nvidia-cuda-nvrtc-cu12 12.5.82
Uninstalling nvidia-cuda-nvrtc-cu12-12.5.82:
Successfully uninstalled nvidia-cuda-nvrtc-cu12-12.5.82
Attempting uninstall: nvidia-cuda-cupti-cu12
Found existing installation: nvidia-cuda-cupti-cu12 12.5.82
Uninstalling nvidia-cuda-cupti-cu12-12.5.82:
Successfully uninstalled nvidia-cuda-cupti-cu12-12.5.82
Attempting uninstall: nvidia-cublas-cu12
Found existing installation: nvidia-cublas-cu12 12.5.3.2
Uninstalling nvidia-cublas-cu12-12.5.3.2:
Successfully uninstalled nvidia-cublas-cu12-12.5.3.2
Attempting uninstall: nvidia-cusparses-cu12
Found existing installation: nvidia-cusparses-cu12 12.5.1.3
Uninstalling nvidia-cusparses-cu12-12.5.1.3:
Successfully uninstalled nvidia-cusparses-cu12-12.5.1.3
Attempting uninstall: nvidia-cudnn-cu12
Found existing installation: nvidia-cudnn-cu12 9.3.0.75
Uninstalling nvidia-cudnn-cu12-9.3.0.75:
Successfully uninstalled nvidia-cudnn-cu12-9.3.0.75
Attempting uninstall: nvidia-cusolver-cu12
Found existing installation: nvidia-cusolver-cu12 11.6.3.83
Uninstalling nvidia-cusolver-cu12-11.6.3.83:
Successfully uninstalled nvidia-cusolver-cu12-11.6.3.83
Successfully installed nvidia-cublas-cu12-12.4.5.8 nvidia-cuda-cupti-cu12-12.4.127 nvidia-cuda-nvrtc-cu12-12.4.127 nvidia-cuda-runtime-cu12-12.4.127 nvidia-cufft-cu12-11.2.3.61 nvidia-curand-cu12-10.3.6.82 nvidia-cusparses-cu12-12.5.1.3 nvidia-cusolver-cu12-11.6.3.83 nvidia-cudnn-cu12-9.3.0.75 nvidia-nvjitlink-cu12-12.5.82 watchdog-6.0.0 streamlit-1.45.1 pydeck-0.9.1

```

```
from transformers import pipeline
```

```
summarizer = pipeline("summarization", model="sshleifer/distilbart-cnn-12-6")
```

```

/usr/local/lib/python3.11/dist-packages/huggingface_hub/utils/_auth.py:94: UserWarning:
The secret `HF_TOKEN` does not exist in your Colab secrets.
To authenticate with the Hugging Face Hub, create a token in your settings tab (https://huggingface.co/settings/tokens), set it as secret
You will be able to reuse this secret in all of your notebooks.
Please note that authentication is recommended but still optional to access public models or datasets.
warnings.warn(
config.json: 100% 1.80k/1.80k [00:00<00:00, 151kB/s]
pytorch_model.bin: 100% 1.22G/1.22G [00:07<00:00, 199MB/s]
model.safetensors: 10% 126M/1.22G [00:01<00:09, 118MB/s]
tokenizer_config.json: 100% 26.0/26.0 [00:00<00:00, 1.72kB/s]
vocab.json: 100% 899k/899k [00:00<00:00, 5.60MB/s]
merges.txt: 100% 456k/456k [00:00<00:00, 7.02MB/s]
Device set to use cpu

```

```

incident_text = """
On June 5, 2024, Unit 3 experienced a temporary loss of power due to a failed generator control module.
Operators responded immediately and shifted cooling operations to backup systems.
No radiation was released. The NRC was notified, and full power was restored within 3 hours.
Investigation revealed that the failure was due to aged circuit components.
"""

```

```

summary = summarizer(incident_text, max_length=100, min_length=30, do_sample=False)
print(summary[0]['summary_text'])

```

⚠ Your max_length is set to 100, but your input_length is only 78. Since this is a summarization task, where outputs shorter than the input are possible, the output may be shorter than expected.

Unit 3 experienced a temporary loss of power due to a failed generator control module . Operators responded immediately and shifted cooling operations to backup systems.

```

from transformers import BartTokenizer, BartForConditionalGeneration

```

```

tokenizer = BartTokenizer.from_pretrained("facebook/bart-large-cnn")
model = BartForConditionalGeneration.from_pretrained(
    "facebook/bart-large-cnn",
    attn_implementation="eager"
)

```

```

incident_text = """
On June 5, 2024, Unit 3 experienced a temporary loss of power due to a failed generator control module.
Operators responded immediately and shifted cooling operations to backup systems.
No radiation was released. The NRC was notified, and full power was restored within 3 hours.
Investigation revealed that the failure was due to aged circuit components.
"""

```

```

inputs = tokenizer(incident_text, return_tensors="pt", truncation=True, max_length=1024)
output = model.generate(
    **inputs,
    max_length=50,
    min_length=20,
    num_beams=4,
    early_stopping=True,
    output_attentions=True,
    return_dict_in_generate=True
)

```

```

summary_text = tokenizer.decode(output['sequences'][0], skip_special_tokens=True)
print("Summary:", summary_text)

```

⚠ Unit 3 experienced a temporary loss of power due to a failed generator control module. No radiation was released. Investigation revealed that the failure was due to aged circuit components.

```

# Cross-attention from last decoder layer (list: layers → batch → heads → target → source)
cross_attn = output.cross_attentions[-1][0] # shape: [num_heads, target_len, source_len]

```

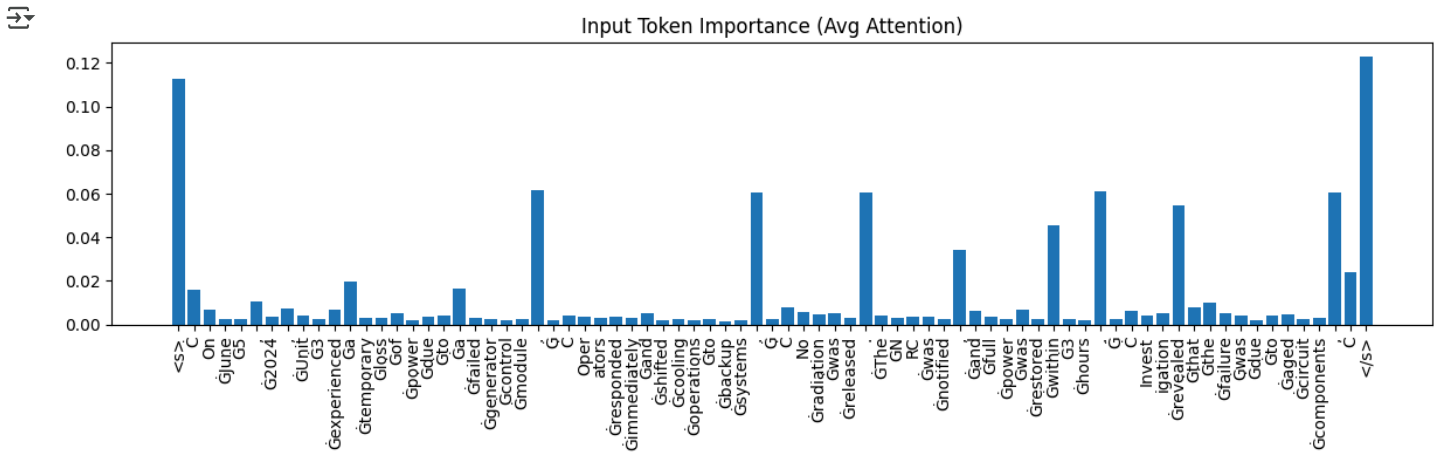
```

import numpy as np
import matplotlib.pyplot as plt

```

```
# Average across heads and target tokens
avg_attn = cross_attn.mean(dim=0).mean(dim=0).detach().numpy().flatten() # shape: [source_len]
tokens = tokenizer.convert_ids_to_tokens(inputs['input_ids'][0])
```

```
plt.figure(figsize=(12, 4))
plt.bar(range(len(tokens)), avg_attn)
plt.xticks(range(len(tokens)), tokens, rotation=90)
plt.title("Input Token Importance (Avg Attention)")
plt.tight_layout()
plt.show()
```



Start coding or [generate](#) with AI.