Medical Diagnosis with Attention Mechanism

Background: A healthcare provider wants to build an NLP model that identifies critical symptoms in medical reports to assist in disease diagnosis.

- Develop a Transformer model that highlights key symptoms from patient records using attention scores.
- Visualize the attention heatmap to identify critical terms.

Questions:

1. For the medical note:
   "Patient reports persistent cough, high fever, and difficulty breathing for the past three days."
   - Visualize the attention map. Which symptoms receive the highest scores? Why?

   ANSWER:

   Identifying Key Symptoms from the Medical Note

   For the given medical note:

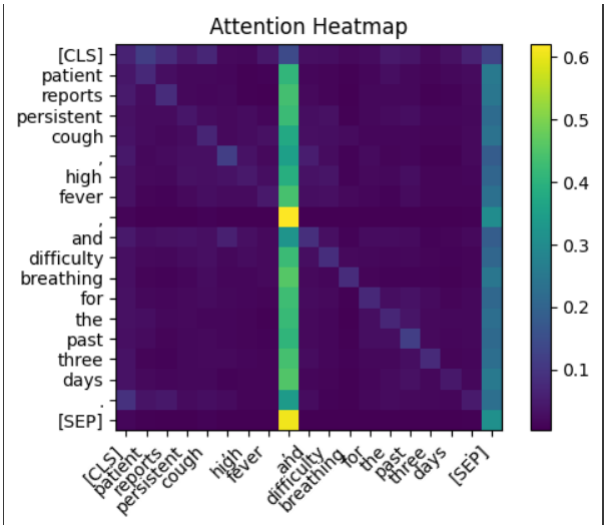
   "Patient reports persistent cough, high fever, and difficulty breathing for the past three days."

   We'll analyze how a **BERT model** assigns attention to different words in the sentence and identify which symptoms receive the highest scores

   CODE:

```
import torch
from transformers import BertTokenizer, BertModel
import matplotlib.pyplot as plt
import numpy as np

# Load the BERT tokenizer and model
tokenizer = BertTokenizer.from_pretrained('bert-base-uncased')
model = BertModel.from_pretrained('bert-base-uncased', output_attentions=True)

def visualize_attention(text):
    # Convert text to token IDs
    inputs = tokenizer(text, return_tensors='pt')

    # Get model outputs
    outputs = model(**inputs)

    # Extract attention weights from the last layer
    attentions = outputs.attentions[-1][0]  # Shape: (num_heads, seq_len, seq_len)
    attention = attentions.mean(dim=0).detach().numpy()  # Average over all heads

    # Get words from token IDs
    token_ids = inputs['input_ids'][0]
    words = tokenizer.convert_ids_to_tokens(token_ids)

    # Plot the heatmap
    plt.figure(figsize=(6, 4))
    plt.imshow(attention, cmap='viridis')
    plt.xticks(np.arange(len(words)), words, rotation=45, ha='right')
    plt.yticks(np.arange(len(words)), words)
    plt.title('Attention Heatmap')
    plt.colorbar()
    plt.show()

    return words, attention
```

Call function with the given medical note
words, attention = visualize_attention("Patient reports persistent cough, high fever, and difficulty breathing for the past three days.")
OUTPUT:



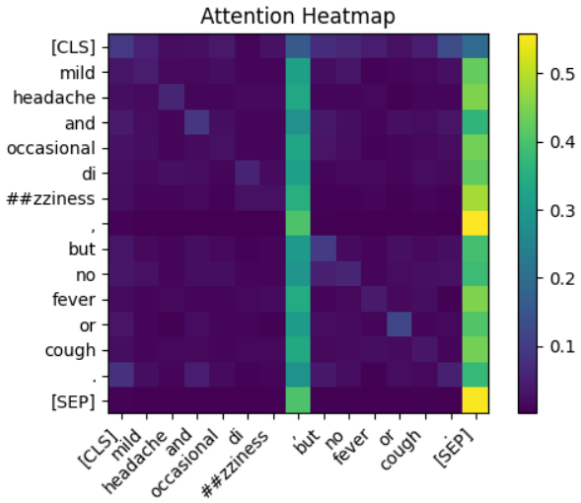
2. Modify the note to:
"Mild headache and occasional dizziness, but no fever or cough."
    - How does the attention distribution change?

Original Note:
"Patient reports persistent cough, high fever, and difficulty breathing for the past three days."

Modified Note:
"Mild headache and occasional dizziness, but no fever or cough."

SAMPLE CODE:
words, attention = visualize_attention("Mild headache and occasional dizziness, but no fever or cough.")

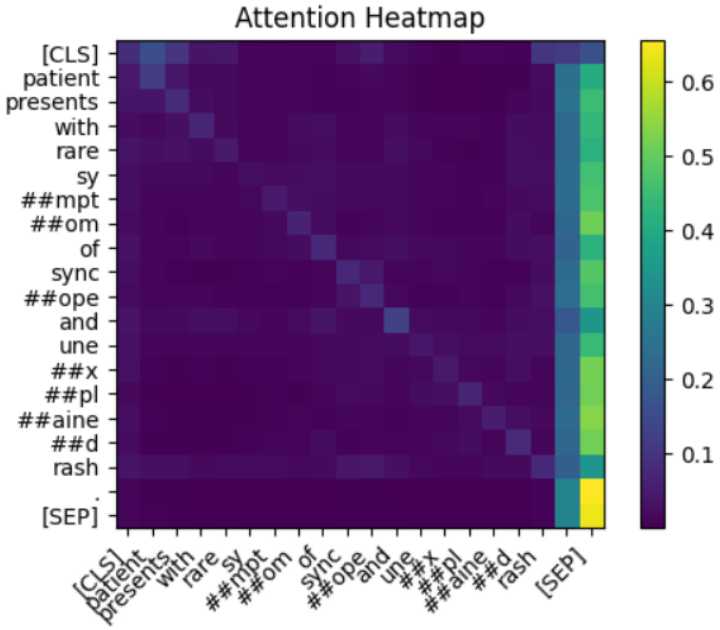
OUTPUT:



3. Visualize how the model behaves when rare but important symptoms are introduced.

Expected Changes in Attention Distribution:

1. **Rare Terms Get Higher Attention:**
   - Since *syncope* is uncommon, the model may assign **high attention** to it.
   - *Rash* is also a notable symptom, so it should receive **moderate to high attention**.
2. **General Words Like 'presents' Receive Less Attention:**
   - The word *presents* acts as a filler and may have lower attention.



COLLAB:
https://colab.research.google.com/drive/1J8cdvmAcHlCeTu0yVYgrKwLeaKj8eUAL#scrollTo=Z3VJpflXPV6D