Exploring Transformer Architecture

An Interactive Lab Using the *Transformer Explainer*

`https://poloclub.github.io/transformer-explainer/`

Instructions

Open your browser and navigate to website above. This interactive tool visualizes how GPT-2, a transformer-based language model, processes text. Work through each section below *in order*, interacting with the site as directed before answering the questions. Take your time exploring — click on components, hover over elements, and read pop-up explanations!

Estimated Time: 60–75 minutes **Points:** _____ / 80

Part 1: Overview and Big Picture (10 pts)

Instructions

When the page loads, you will see an animated overview. Read the introduction text (scroll down) then watch the animation for at least one full cycle before answering.

Question 1.1 — What Is a Transformer? (3 pts)

In your own words, describe what a transformer model does. What is it trying to *predict*?

Question 1.2 — High-Level Components (4 pts)

List the **four major stages** you can identify in the top-level diagram. (Hint: follow the data flow from input to output.)

1. _____
2. _____
3. _____
4. _____

Question 1.3 — Interactive Input (3 pts)

Type a short phrase into the text input box on the site (e.g., *"Data science is"*). What phrase did you enter, and what token did the model predict as the next word?

Your input phrase: _____

Predicted next token: _____

Part 2: Tokenization and Embeddings (20 pts)

Key Concept

Transformers cannot read raw text. Text must first be broken into **tokens** and converted into numerical **vectors** (embeddings) that capture semantic meaning.

Question 2.1 — Tokenization (5 pts)

Click on the **Tokenization** step. Using the phrase you entered in Q1.3, list each token the model creates and its corresponding token ID.

Token (text)	Token ID

Why do you think some words are split into multiple tokens?

Question 2.2 — Token Embeddings (5 pts)

Explore the **Token Embedding** component. What is the dimensionality (size) of each token embedding vector in GPT-2? Why do you think a high-dimensional vector is useful for representing a word?

Embedding dimension: _____

Explanation:

Question 2.3 — Positional Encoding (5 pts)

Transformers process all tokens simultaneously. Why is **positional encoding** necessary? What problem does it solve?

Question 2.4 — Combining Embeddings (5 pts)

After token embeddings and positional encodings are created, how are they combined into a single representation?

Part 3: Attention Mechanism (20 pts)

Key Concept

The **self-attention** mechanism allows each token to "look at" other tokens in the sequence and decide which ones are most relevant for understanding its meaning. This is the core innovation of the transformer.

Question 3.1 — Attention Scores (6 pts)

Hover over the attention score visualization. Pick any one token from your input phrase and describe which other token(s) it attends to most strongly. Does this relationship make intuitive sense? Explain.

Token examined: _____

Attends most to: _____

Does it make sense? Why or why not?

Question 3.2 — Softmax and Normalization (4 pts)

After computing raw attention scores, a **softmax** function is applied (each word corresponds to a row in the array). Choose a word, and find which word corresponds to the largest value in the softmax array.

Question 3.4 — Multi-Head Attention (6 pts)

GPT-2 uses **multi-head attention** rather than a single attention function. How many attention heads does the visualization show? What is the advantage of having multiple heads?

Number of attention heads: _____

Advantage of multiple heads:

Question 3.5 — Causal (Masked) Attention (4 pts)

Notice that GPT-2 uses **masked self-attention**. What does masking do (Hint: See Slide 11/20)?

Part 4: Feed-Forward Network and Layer Norm (10 pts)

Question 4.1 — Feed-Forward Sublayer (5 pts)

After attention, each token passes through a **Feed-Forward Network (FFN)**. What role does the FFN play in the overall model? (Slide 13)

Question 4.2 — Activation Function (5 pts)

Identify the **activation function** used inside the FFN. What is the purpose of a non-linear activation function in a neural network? (Slide 13)

Activation function used: _____

Purpose of non-linearity:

Part 5: Output and Probability Distribution (10 pts)

Question 5.1 — Temperature (5 pts)

Experiment with the **temperature** slider on the site (if available) or read its description. How does changing the temperature affect the model's output distribution? What happens at very high vs. very low temperatures?

Question 5.2 — Top Predictions (5 pts)

Look at the top predicted tokens for your phrase. List the **top 3 predicted tokens** and their probabilities. Do the predictions seem reasonable? Why or why not?

Token	Probability

Are predictions reasonable?

Part 6: Synthesis and Reflection (10 pts)

Question 6.1 — Open Reflection (10 pts)

Describe **one thing that surprised you** about how transformers work, and **one question** you still have after completing this lab.

Something that surprised me:

A question I still have:

Transformer Explainer was created by Polo Club of Data Science at Georgia Tech.
https://poloclub.github.io/transformer-explainer/

This lab was partially generated by Claude AI.