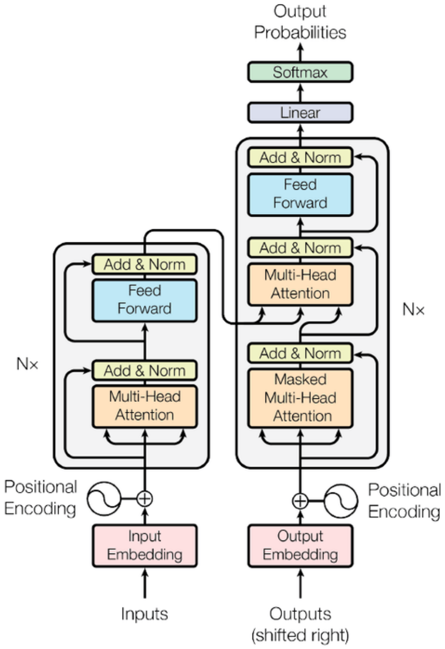

This diagram is the architecture of the **Transformer model** used in deep learning, especially for natural language processing (like in ChatGPT). It has two main parts: **Encoder** (left) and **Decoder** (right).

Here is a **step-by-step explanation**:

🔵 **ENCODER (Left side)**

1. **Input Embedding**:
   - Convert each input word into a vector using an embedding layer.
2. **Positional Encoding**:
   - Add position information to each word vector since transformers don't know order by themselves.
3. **Stack of N Encoder Layers**:
   Each encoder layer has:
   - **Multi-Head Attention**: Focuses on different parts of the input sentence at the same time.
   - **Add & Norm**: Adds the attention output back to the input and normalizes it.
   - **Feed Forward**: A fully connected layer that transforms the data.
   - **Add & Norm** again.

4. This entire encoder stack is repeated **N times** (usually 6 in the original transformer).

🟠 DECODER (Right side)

1. **Output Embedding (shifted right)**:
   - Similar to input embedding, but for output words. Shifted right to prevent using future words.
2. **Positional Encoding**:
   - Add position info to output word vectors.
3. **Stack of N Decoder Layers**:
   Each decoder layer has:
   - **Masked Multi-Head Attention**: Looks at previous output tokens but **not future tokens** (prevents cheating).
   - **Add & Norm**
   - **Multi-Head Attention** (with encoder output): Connects the decoder to the encoder output (helps decoder understand the input).
   - **Add & Norm**
   - **Feed Forward**: Transforms the data.
   - **Add & Norm**
4. This decoder stack is repeated **N times**.

🟢 FINAL STEP

5. **Linear + Softmax Layer**:
   - Convert the decoder's output into probabilities for each possible word.
   - The highest probability word is chosen as the next word in the output.

Summary in Short Steps:

1. Input → Embedding + Position
2. Encoder → N layers (Attention + FeedForward)
3. Output → Embedding + Position (Shifted)
4. Decoder → N layers (Masked Attention + Encoder Attention + FeedForward)

5. Final → Linear → Softmax → Output word probabilities