

Hide menu

Text Summarization

- ✔

**Video:** Week Introduction

53 sec
- ✔

**Video:** Transformers vs RNNs

3 min
- ✔

**Reading:** Transformers vs RNNs

10 min
- ✔

**Video:** Transformers overview

5 min
- ▶

**Video:** Transformer Applications

7 min
- ✔

**Reading:** Transformer Applications

10 min
- ▶

**Video:** Scaled and Dot-Product Attention

3 min
- ▶

**Video:** Masked Self Attention

3 min
- ▶

**Video:** Multi-head Attention

5 min
- Ⓜ

**Reading:** Multi-head Attention

10 min
- ✔

**Lab:** Attention

1h
- ✔

**Lab:** Masking

1h
- ✔

**Lab:** Positional encoding

1h
- ▶

**Video:** Transformer Decoder

4 min
- Ⓜ

**Reading:** Transformer Decoder

10 min
- ▶

**Video:** Transformer Summarizer

4 min
- ▶

**Video:** Week Conclusion

34 sec
- Ⓜ

**Reading:** Content Resource

10 min

Lecture Notes (Optional)

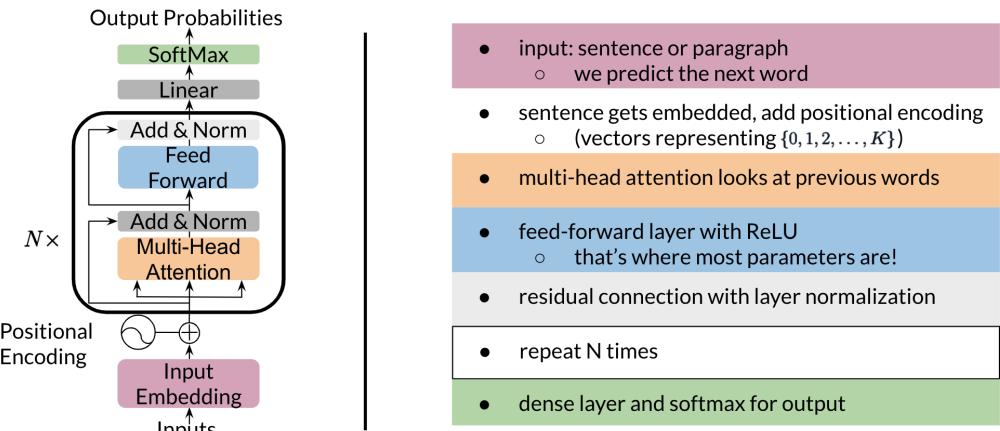
Practice Quiz

Assignment

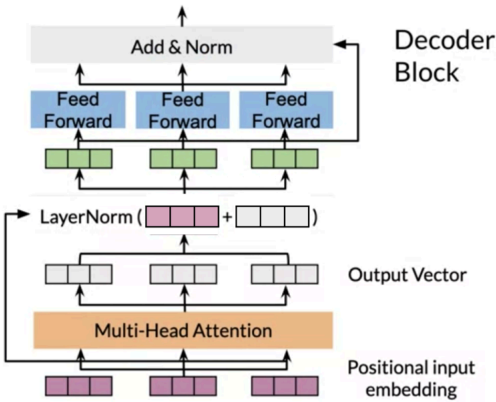
[🏠](#) > [Week 2](#) > Transformer Decoder

[< Previous](#) [Next >](#)

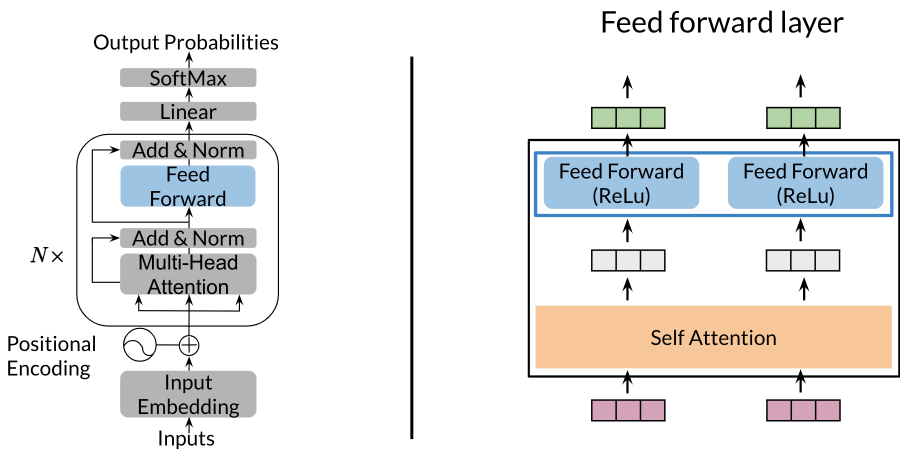
That was a lot of information! Don't worry if you did not understand everything, we will go over everything step by step.



Once you get the embeddings, you add the positional encoding, which you can think of just a number that tells you information about the position of the word. Then, you do multi-head attention as explained in the previous video/reading. There is a feedforward layer (blue) with a ReLU (and optionally some linear layers) after this, then a residual connection with layer normalization (repeat the block shown above N times), finally a linear layer with a softmax. Zooming into the decoder block that gets repeated N times, you get the following:



Now for the feedforward block, you can just implement the following:



You get the input, (red vector) run it through self-attention and then a feedforward with ReLU. At the end of the decoder, you can just run a linear layer and a softmax.