Text Summarization

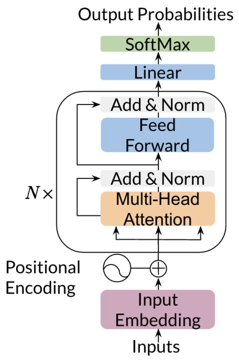
- ✓ **Video:** Transformers vs RNNs 6 min
- ✓ **Reading:** Transformers vs RNNs 10 min
- ✓ **Video:** Transformer Applications 8 min
- ✓ **Reading:** Transformer Applications 10 min
- ✓ **Video:** Dot-Product Attention 7 min
- ✓ **Reading:** Dot-Product Attention 10 min
- ✓ **Video:** Causal Attention 4 min
- ✓ **Reading:** Causal Attention 10 min
- ✓ **Video:** Multi-head Attention 6 min
- ✓ **Reading:** Multi-head Attention 10 min
- **Lab:** Attention 1h
- ✓ **Video:** Transformer Decoder 5 min
- ✓ **Reading:** Transformer Decoder 10 min
- ✓ **Video:** Transformer Summarizer 4 min
- **Reading:** Transformer Summarizer 10 min
- **Lab:** The Transformer Decoder 1h
- **Reading:** Content Resource 10 min

Assignment

- **Programming Assignment:** Transformer Summarizer 3h

Transformer Summarizer

You will be building a transformer summarizer this week. To process your data, you will need to process your text as follows:



Model Input:

```
ARTICLE TEXT <EOS> SUMMARY <EOS> <pad> ...
```

Tokenized version:

```
[2,3,5,2,1,3,4,7,8,2,5,1,2,3,6,2,1,0,0]
```

Loss weights: 0s until the first `<EOS>` and then 1 on the start of the summary.

You keep generating words by random sampling until you get the end-of-sentence (EOS) token. With this, it's possible that you get a different summary each time you run the model. The loss function you will be using in the programming assignment is a simple cross entroy loss function (described in course 3).

Mark as completed