NLP Project

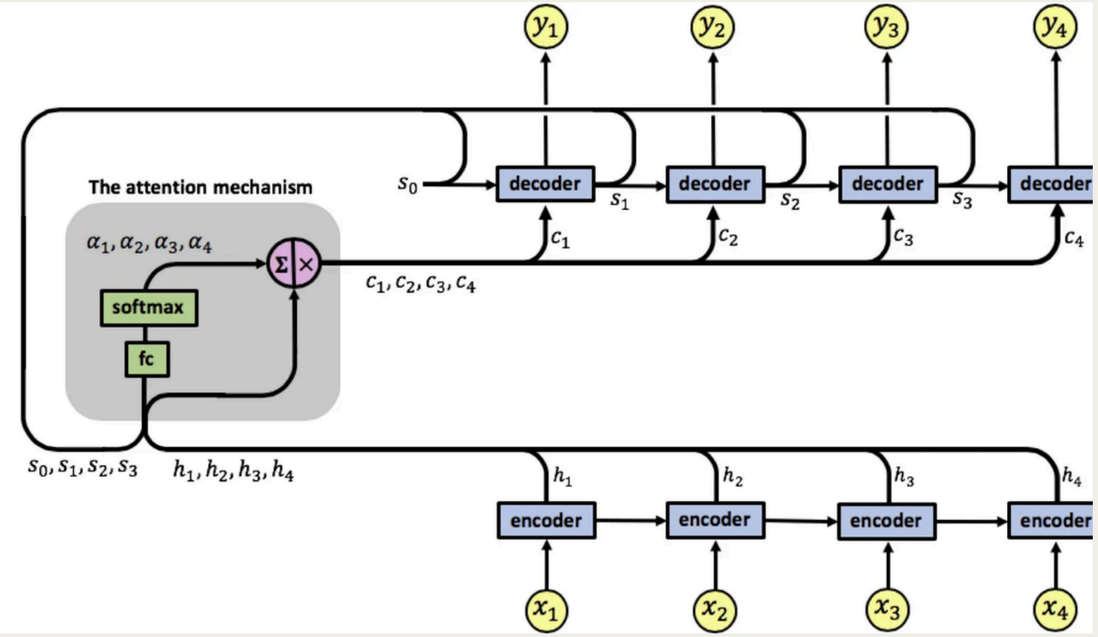
HEADLINE GENERATION

Aryan Tripathi 202101036
Dev Patel 202101092
Aaditya Meher 202103024

LSTM ENCODER-DECODER MODEL WITH ATTENTION

Encoder

- One or more LSTM (Long Short-Term Memory) layers.
- Repeatedly takes in word embedding and previous hidden state and compute next hidden state



Decoder

- One or more LSTM layers.
- Takes in previous hidden state, previous output and attention to produce next hidden state
- Generate till end-of-sequence token or maximum length reached

Attention Mechanism

- The model uses attention to focus on different parts of the input sequence when generating tokens in the output sequence.
- At each time step, an attention mechanism computes a weighted attention vector, based on the relevance of each encoder hidden state to the current decoding step using a neural network.
- This dynamic approach allows the model to attend to different input parts.

TRANSFORMER MODEL

Self-Attention Mechanism

- The self-attention mechanism calculates attention scores at each position in an input sequence, which are determined using learnable parameters like query, key, and value vectors, and transformed into attention weights using a softmax function.
- The final output is a weighted sum of all positions' values.

Multi-head Attention

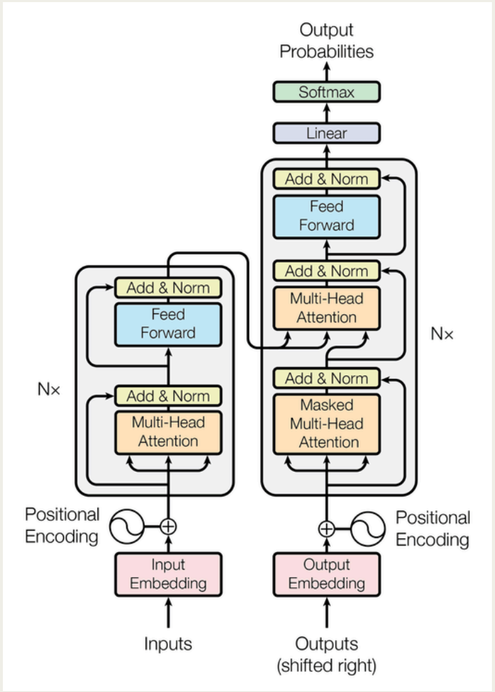
- The model's self-attention mechanism is expanded to include multiple attention heads, each with its own learnable parameters.
- During training, the decoder uses masked multi-head attention to prevent it from focusing on future tokens.

Feed Forward Neural Network

- We use a fully connected neural network to learn complex input-output mapping, transform the representations to a larger space to extract higher level features along with residual connections and layer normalisation

Encoder-Decoder Architecture

The encoder processes the input sequence, while the decoder generates the output sequence.



Output Projection

The output of the decoder layers is passed through a projection and a softmax layer to generate a probability distribution over the vocabulary for each position in the output sequence.

JUSTIFICATION FOR CHOOSING THE MODEL

- First we did naïve text highlight generation from the article by only taking frequency of each word and then outputting sentences which have highest score based on the frequency of words in the sentence
- This was flawed as we may miss some important points in article if it’s score was less and we may output lot of unnecessary and large sentences if its score was large.
- So we used LSTM with attention and transformer models so that they could capture long-term dependencies and can focus on important parts of articles rather than entire sentences using the attention mechanism
- We also used the above models so that we could handle variable length inputs
- It can generate new text or phrases that may not be in the input text which was a flaw of the first frequency based highlight generation model
- Transformers also allow for parallelisation with multi-head attention to capture different contexts within the sentence.

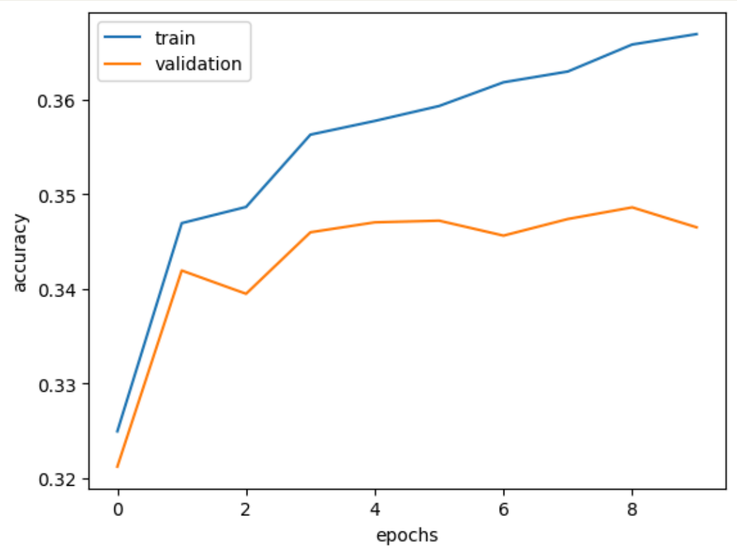
CREATIVITY

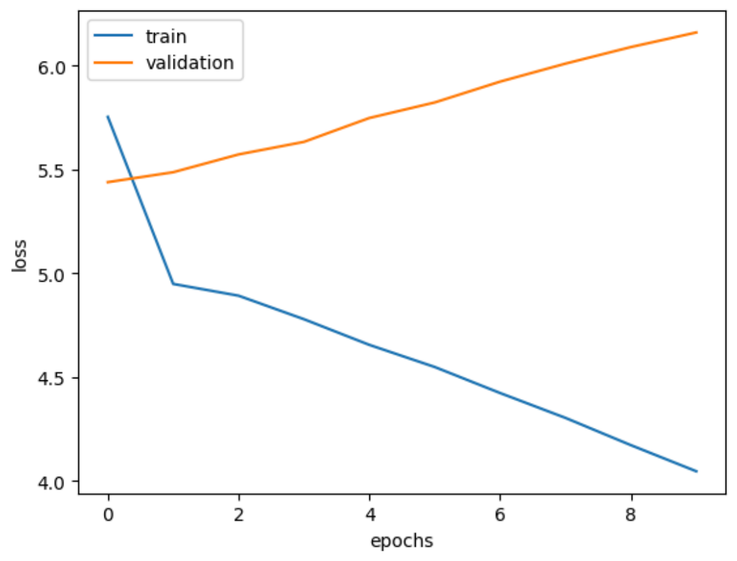
- We employed an encoder-decoder architecture in conjunction with an attention mechanism to extract attention weights associated to the input sequence and construct the current target word.

- Conventional LSTM compresses all previous information in a fixed hidden state. Instead, we applied the attention mechanism, which allowed the model to choose focus on the most relevant sections of the input data, lowering the requirement for information compression.

RESULTS

LSTM





TRANSFORMERS