A Survey on Transfer Learning in NLP

Shannon Phu

Based on *A Survey on Transfer Learning in Natural Language Processing*

Architectures

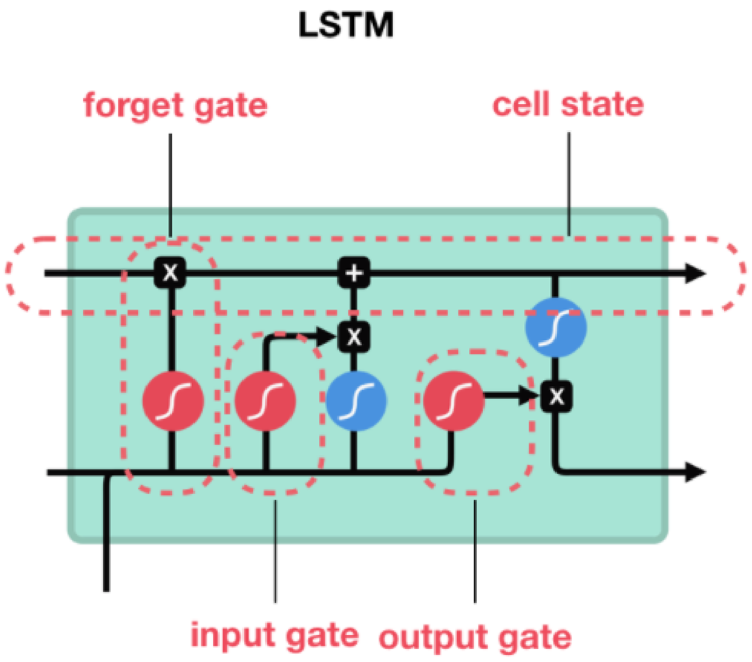
- Recurrent Neural Networks
- LSTM
- Encoder-Decoder
- Attention

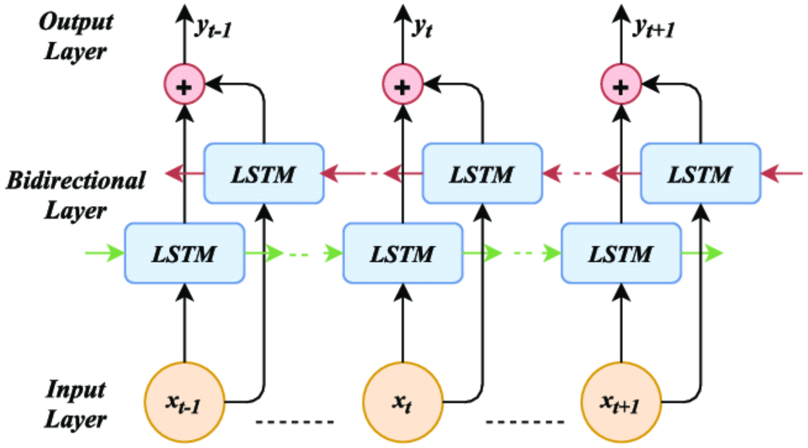
Recurrent Neural Networks

- processes sequential data by passing the previous state along with each input
- Problem:
  - Issues with carrying information from previous steps in a long sequence
  - vanishing gradient problem: backpropagation causes error to decay as the losses are propagated backwards through all the layers causing little to no adjustments in the network weights
  - exploding gradient problem: errors accumulate as they are propagated backwards and cause network weights to overflow

LSTM

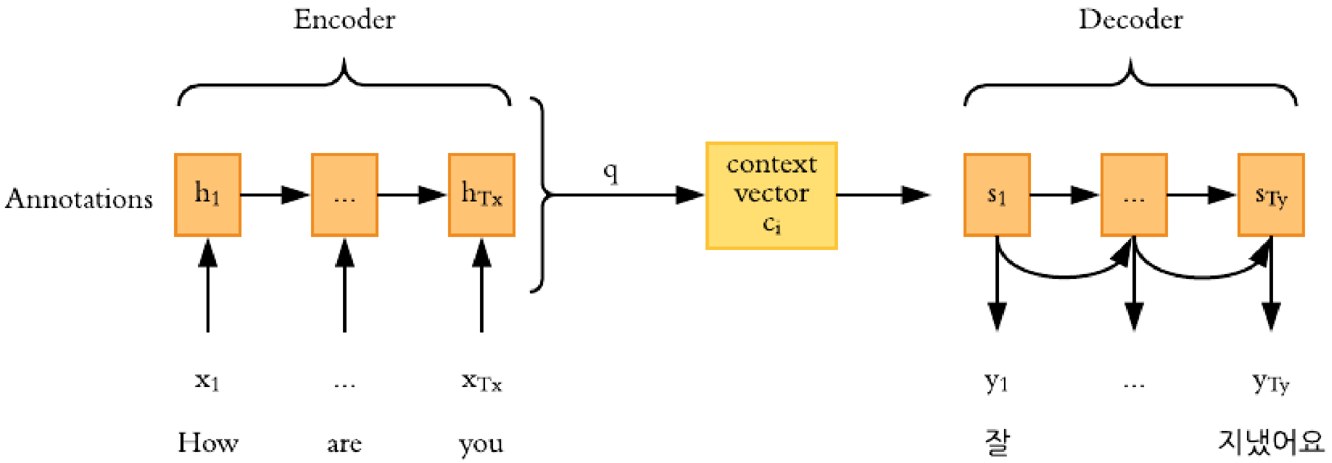
- Long Short Term Memory
- consists of gates and activations which prevent both vanishing and exploding gradients
- layers which can keep long term information or forget anything it doesn't need
- bidirectional LSTM architecture takes advantage of chaining many LSTM units together in both a left-to-right and right-to-left direction





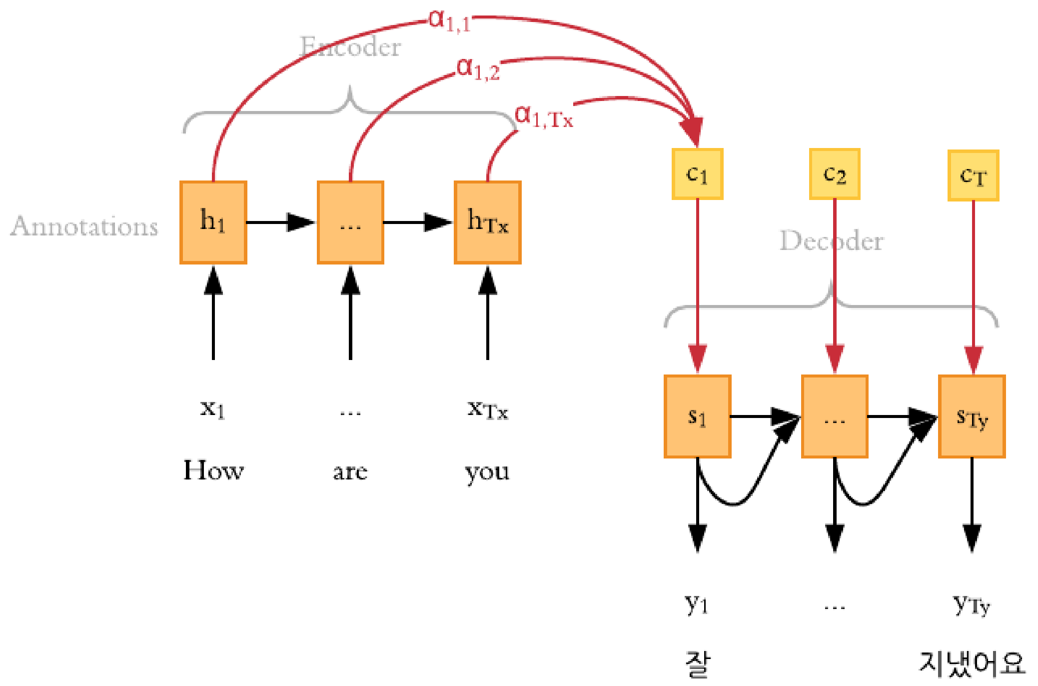
Encoder-Decoder

- sequence to sequence model
- encoder component encodes textual input into a context vector
- decoder component decodes the vector back into the output sequence text



Attention

- encoder remains the same
- decoder on the other hand attends to different parts of the source sentence at each step of the output generation
- hidden state is computed with the context vector, the previous output and the previous hidden state



Types of Transfer Learning

Transductive Transfer Learning

- same task to learn, but the target domain is different from the domain trained on

Inductive Transfer Learning

- different task to learn, but we have labelled data in the target domain

Transfer Learning

- Fine-tuning
  - A pre-trained model's weights will be reused to learn a new task. The original model's parameters at each layer may change
- Feature Embeddings
  - We can learn vector representations for words or sentences and use these as input to a model to learn additional weights for a downstream task.
- Zero-Shot Learning
  - We can also apply a pre-trained model to a brand new task without any further fine-tuning or training and evaluate its performance as-is on the new task. This means we can apply a model to a problem it was not explicitly trained to do.

Examples of Language Models

ULMFiT

- LSTM model with few extra layers for text classification

BERT

- Bidirectional Encoder Representations from Transformers
- transformer-based architecture which learns bidirectional representations of words
- trained on a masked language task and a next sentence prediction task

RoBERTa

- optimizes on top of BERT by training on longer sequences with more data after finding that BERT was undertrained

ELMo

- bidirectional LSTM model which learns deep representations of natural language

GPT-2

- transformer architecture and can perform zero-shot learning on a variety of downstream tasks
- recommended for generative tasks

Conclusion

- Recent advances in natural language processing have continuously improved the state-of-the-art results in common natural language tasks such as question-and-answer, text summarization, translation, and text classification
- There are popular NLP datasets to compare for SOTA evaluation
- From recurrent neural networks, to LSTMs, to the newer transformer architectures, improvements are being made on models to allow for improved model fine-tuning on downstream tasks