

Natural Language Processing

Gobind Puniani
MBG Summer 2020
07/23/20

Sources

- fast.ai NLP course: <https://www.fast.ai/2019/07/08/fastai-nlp/>
- fast.ai documentation: <https://docs.fast.ai/text.html>
- GitHub tutorial on Sentiment Analysis with PyTorch: <https://github.com/bentrevelt/pytorch-sentiment-analysis>
- PyTorch model for Text Classification: https://pytorch.org/tutorials/beginner/text_sentiment_ngrams_tutorial.html
- Explanation of transformers in NLP: <https://www.analyticsvidhya.com/blog/2019/06/understanding-transformers-nlp-state-of-the-art-models/>
- Text classification with Transformer implementation: <https://towardsdatascience.com/https-medium-com-chaturangarajapakshe-text-classification-with-transformer-models-d370944b50ca>

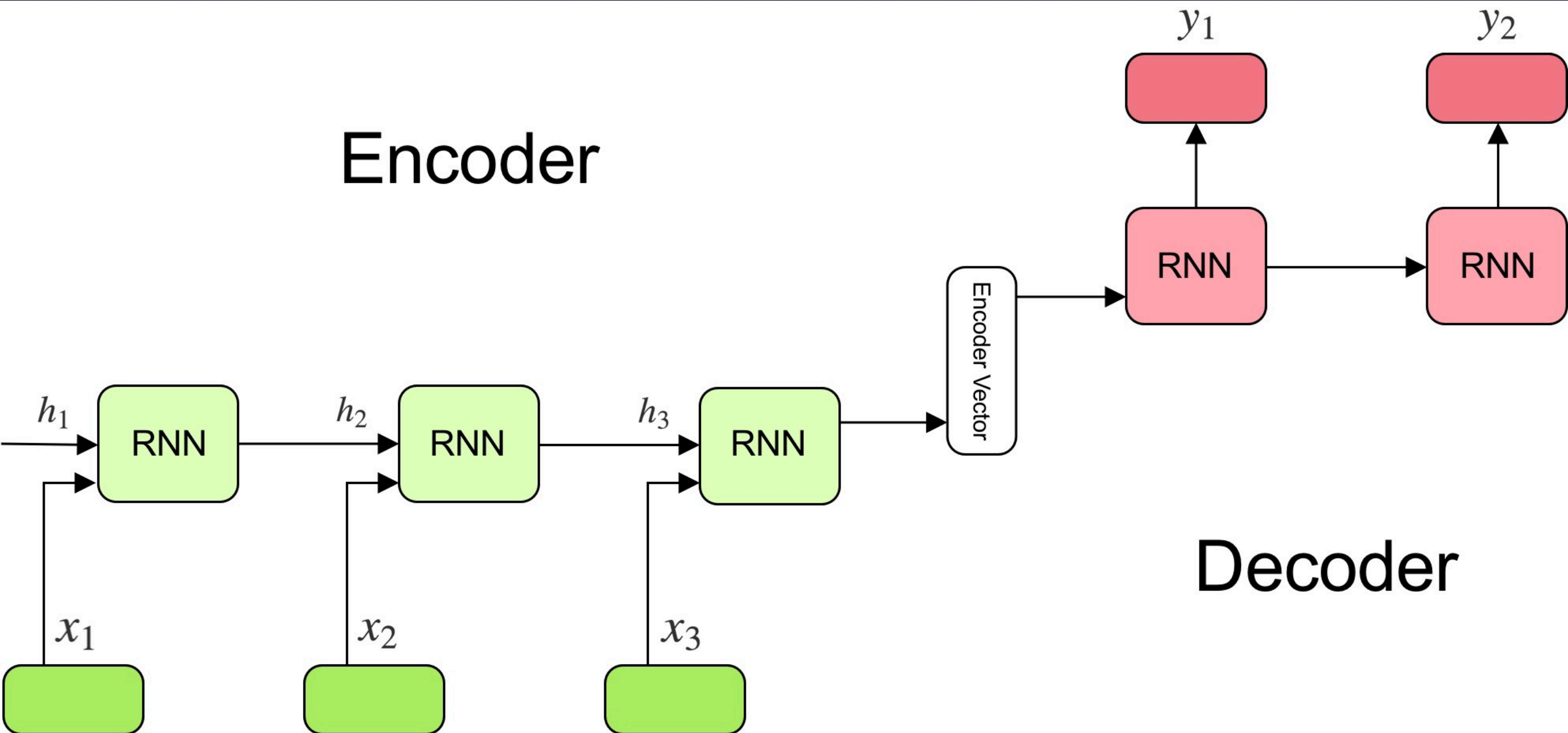
NLP Data

- Text cannot be directly transformed into numbers like images
- Tokenization: format raw texts into lists of words
 - The lists are called “tokens”
- Numericalization: transform the tokens into numbers (ids)
- Numbers are then passed through embedding layers to convert them into arrays to be fed into model

NLP Background

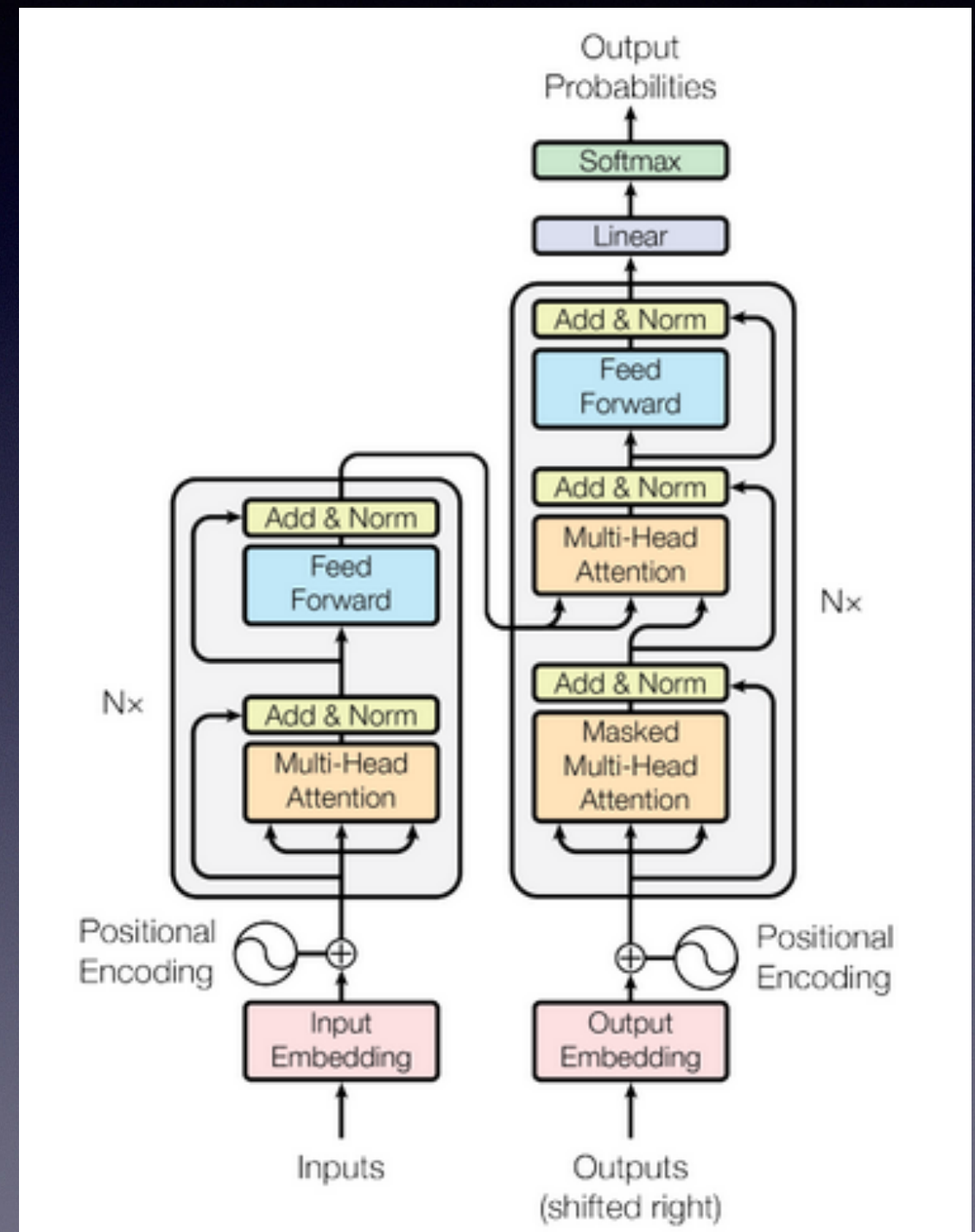
- Currently in the midst of NLP Renaissance
- Sequence-to-sequence (seq2seq) models convert sequences of one type to another
 - E.g. translating from one language to another
- Used in other NLP tasks using RNN basis since 2014
- Attention mechanism introduced in 2015

Seq2Seq Diagram

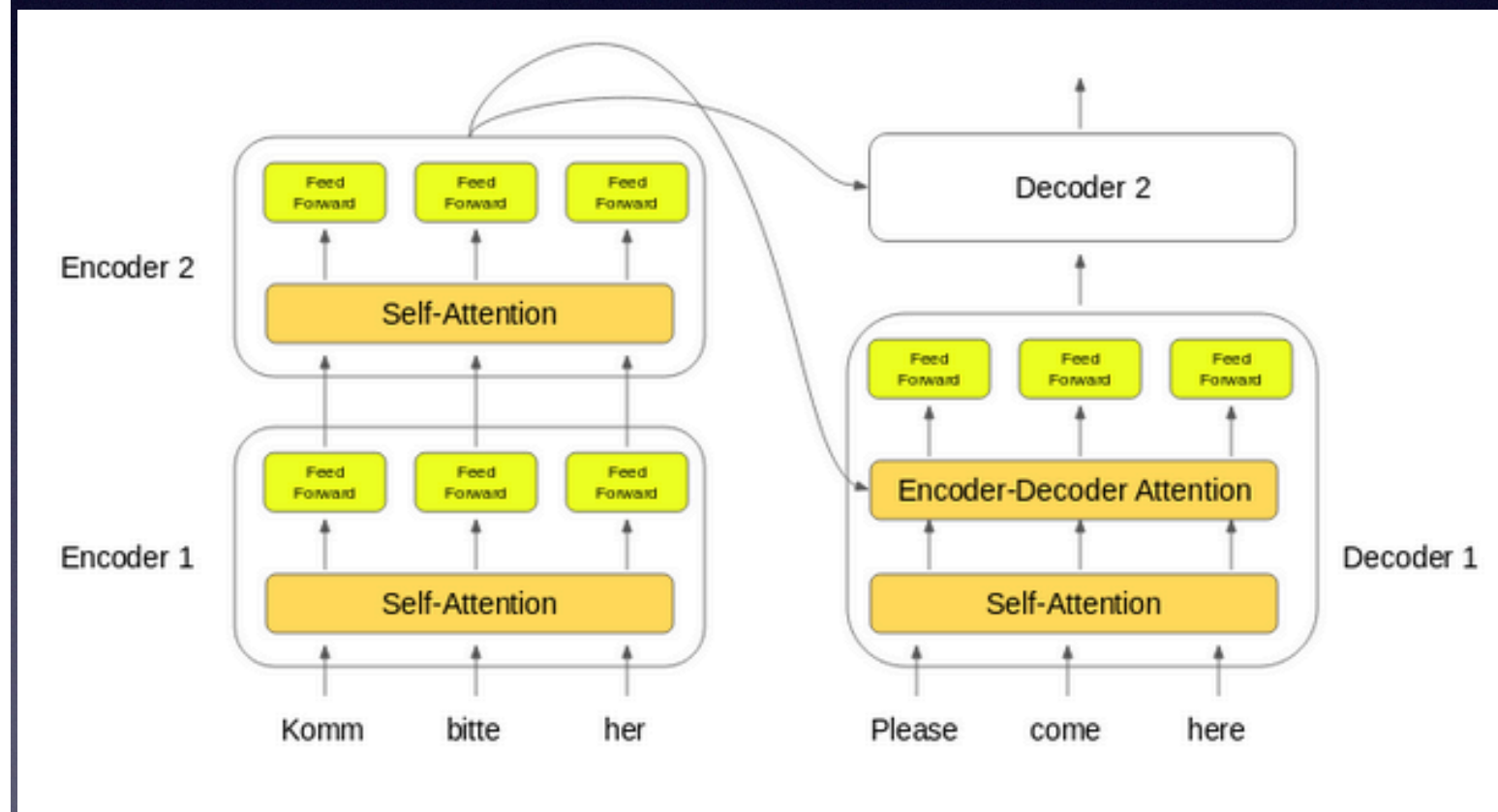
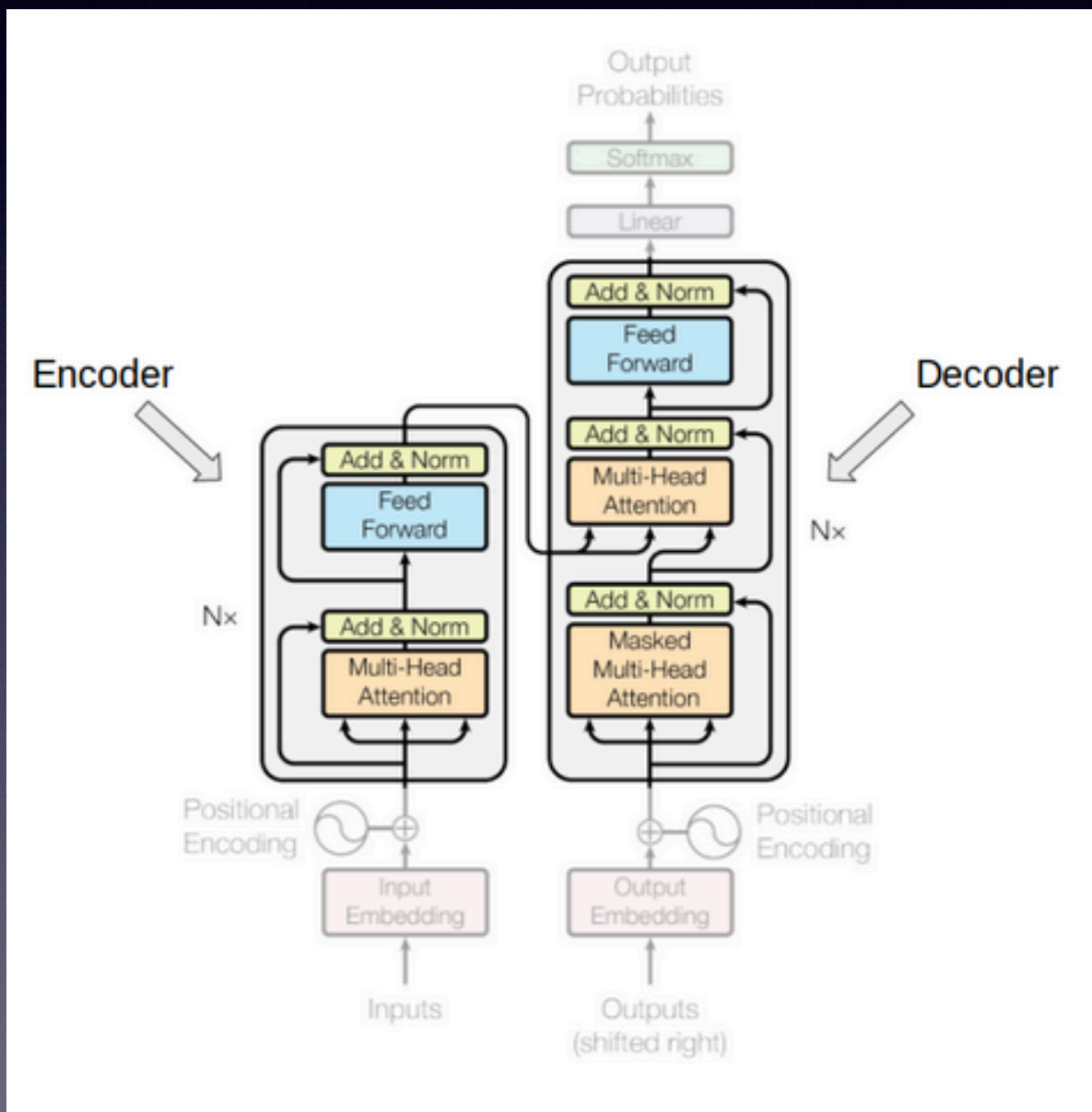


NLP Background (Cont.)

- Problems with seq2seq:
 - Sequential architecture prevents parallelization
 - Issue with long-range dependencies
- Solution: Google Brain's Transformer model

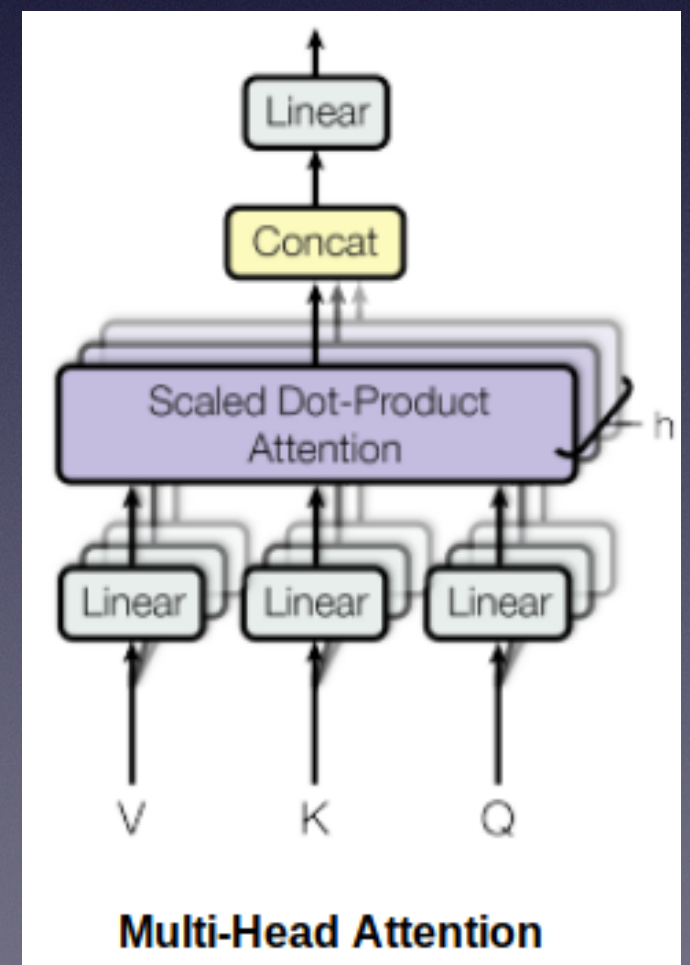
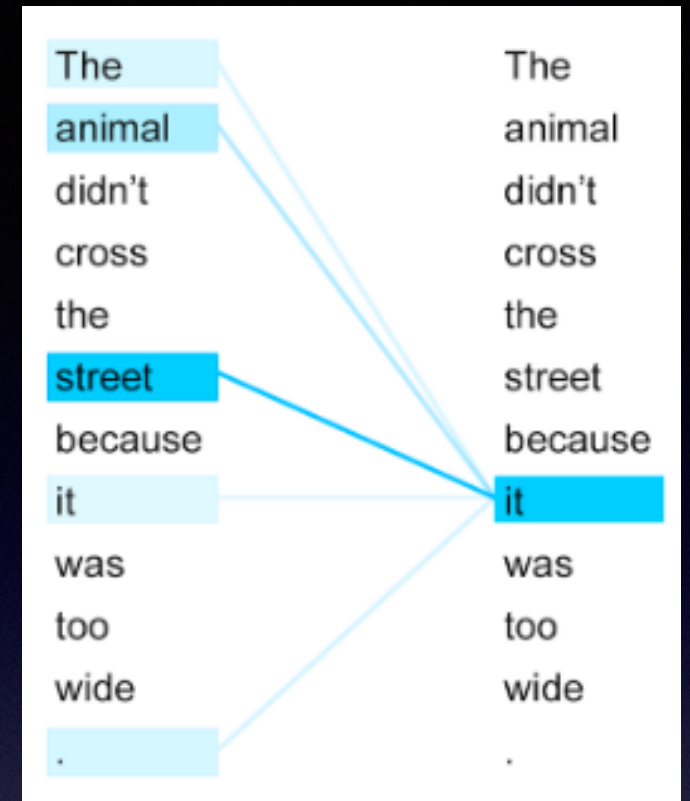


Transformer Diagrams



Attention

- Self-attention (or intra-attention): attention mechanism that examines other words in a sequence to understand a word
- Multi-head attention: self-attention is calculated multiple times in parallel using query vectors, key vectors, and value vectors

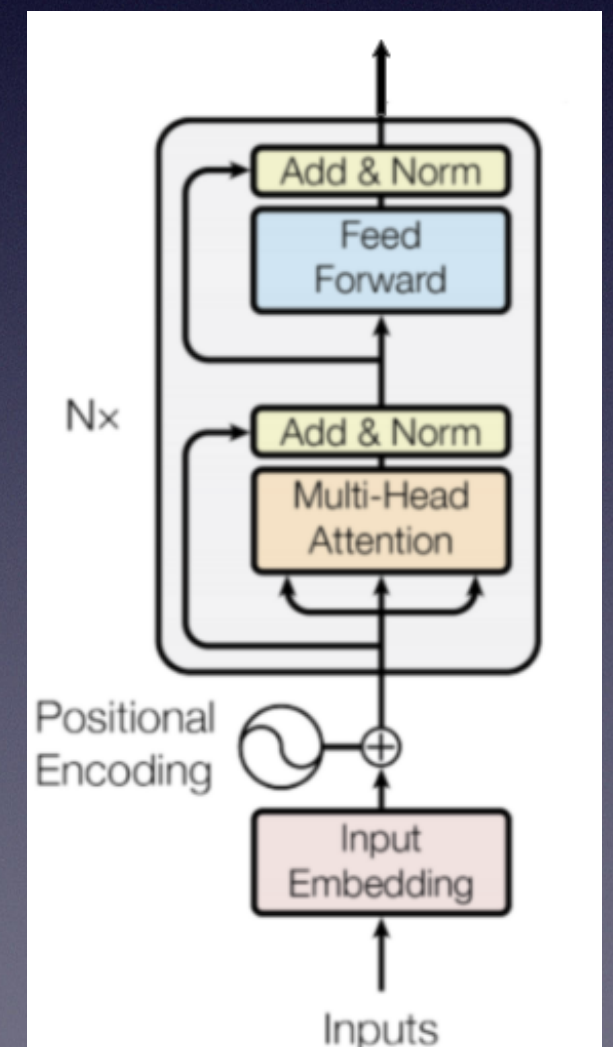
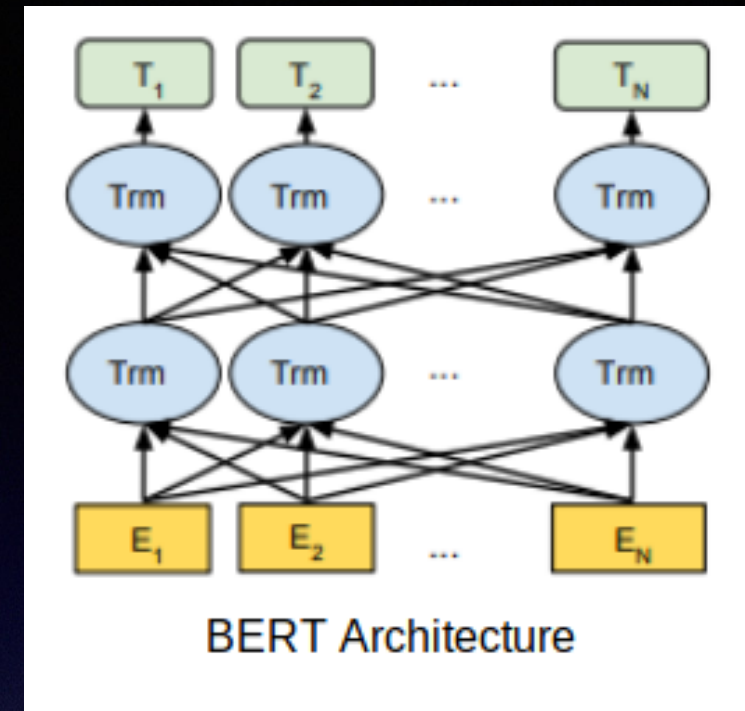


Transformer XL

- Problem: Transformer can only handle fixed-length strings
 - Context fragmentation results from split sentences
 - Attention computation is quadratic to sequence length
- Solution: Transformer XL
 - Re-uses representations from previous segments rather than recomputing them

BERT

- A framework from Google AI released in 2019: Bidirectional Encoder Representations from Transformers
- Uses pre-training and fine-tuning to perform well on several tasks
- Pre-trained on unsupervised Masked Language Modeling and Next Sentence Prediction
- Multi-layer bidirectional Transformer encoder
 - Self-attention layer performs in both directions



Process Overview

- Thoroughly scoured fast.ai, PyTorch, GitHub, Towards Data Science, etc. for implementations
 - Most sidestepped the use of activation functions
- Learner model with fast.ai is not working
- Tried developing own implementation using fast.ai definitions and PyTorch skeleton
 - Unknown model parameters
- Emailed Diganta Misra himself
 - Received link to paper on testing activation functions in NLP

Next Steps

- NLP:
 - Work on Learner version with Manuel
 - Pursue RNN strategy from paper that inspired Misra
- Unsupervised learning with ImageNet or COCO
- Paper:
 - How to include activation functions in tables of results