Contextual Word Embeddings

What's New and How to Use Them?

Shreya Khurana
Data Scientist, GoDaddy

whoami

- Data Scientist at 
- Work with language data and ML models
- Also worked on Bayesian Hierarchical Modeling

Outline

- Introduction to word embeddings
- Word2vec
- GloVe
- ELMo
- How to use word embeddings
- Basic Pre-processing
- Transformer
- BERT
- Other Transformer based models

Vocabulary Growth

Word Embeddings

- Why?
  - Problem of sparsity with one-hot encoding
  - Large vocabulary size: 20k, 32k
  - Encode one word as : $[0, 0,\ldots,1, 0, 0, 0,\ldots 0]_{1x32000}$
  - Space inefficient

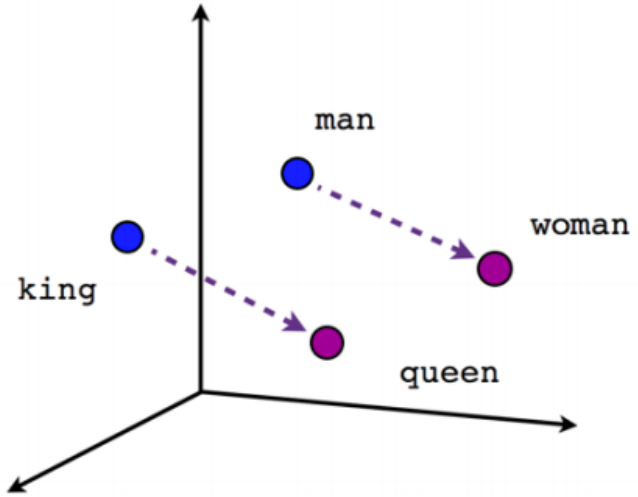
Vocabulary:
Man, woman, boy, girl, prince, princess, queen, king, monarch

		1	2	3	4	5	6	7	8	9
1	man	1	0	0	0	0	0	0	0	0
2	woman	0	1	0	0	0	0	0	0	0
3	boy	0	0	1	0	0	0	0	0	0
4	girl	0	0	0	1	0	0	0	0	0
5	prince	0	0	0	0	1	0	0	0	0
6	princess	0	0	0	0	0	1	0	0	0
7	queen	0	0	0	0	0	0	1	0	0
8	king	0	0	0	0	0	0	0	1	0
9	monarch	0	0	0	0	0	0	0	0	1

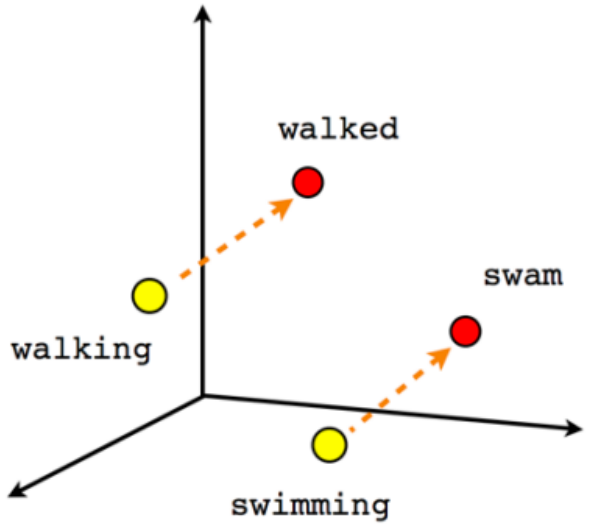
Image source: https://hackernoon.com/various-optimisation-techniques-and-their-impact-on-generation-of-word-embeddings-3480bd7ed54f

Word Embeddings

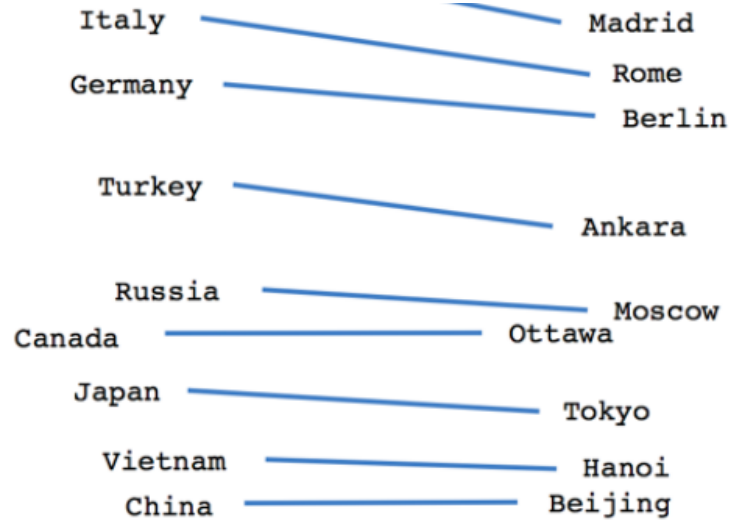
- Representations for text
- Understood by ML models
- Real-valued vectors in a *n*-dimensional space



Male-Female



Verb tense

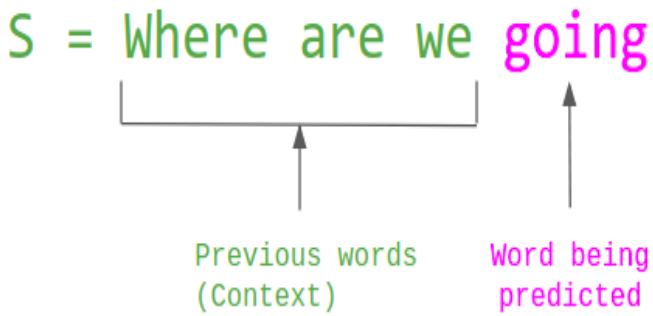


Country-Capital

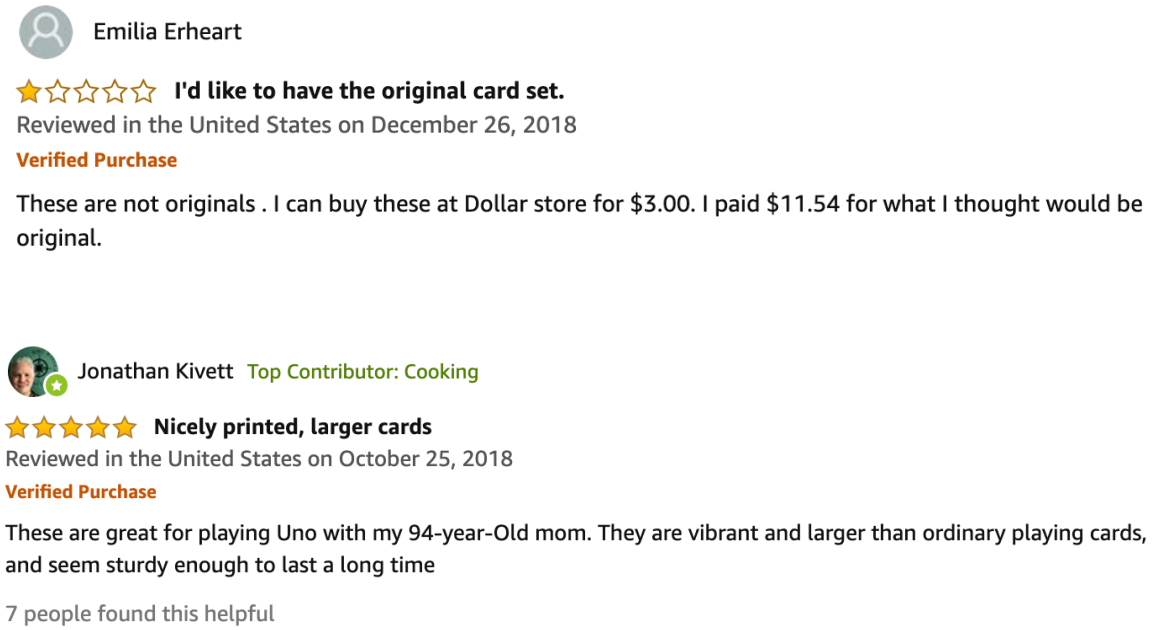
Image source : https://www.analyticsvidhya.com/blog/2017/06/word-embeddings-count-word2veec/

Word Embeddings

- Where are they used?
  - Language modeling
  - Sentiment Analysis
  - Text generation



$$P(S) = P(\text{Where}) \times P(\text{are} \mid \text{Where}) \times P(\text{we} \mid \text{Where are}) \times P(\text{going} \mid \text{Where are we})$$



Emilia Erheart

★☆☆☆☆ **I'd like to have the original card set.**

Reviewed in the United States on December 26, 2018

Verified Purchase

These are not originals . I can buy these at Dollar store for $3.00. I paid $11.54 for what I thought would be original.

Jonathan Kivett Top Contributor: Cooking

★★★★★ **Nicely printed, larger cards**

Reviewed in the United States on October 25, 2018

Verified Purchase

These are great for playing Uno with my 94-year-Old mom. They are vibrant and larger than ordinary playing cards, and seem sturdy enough to last a long time

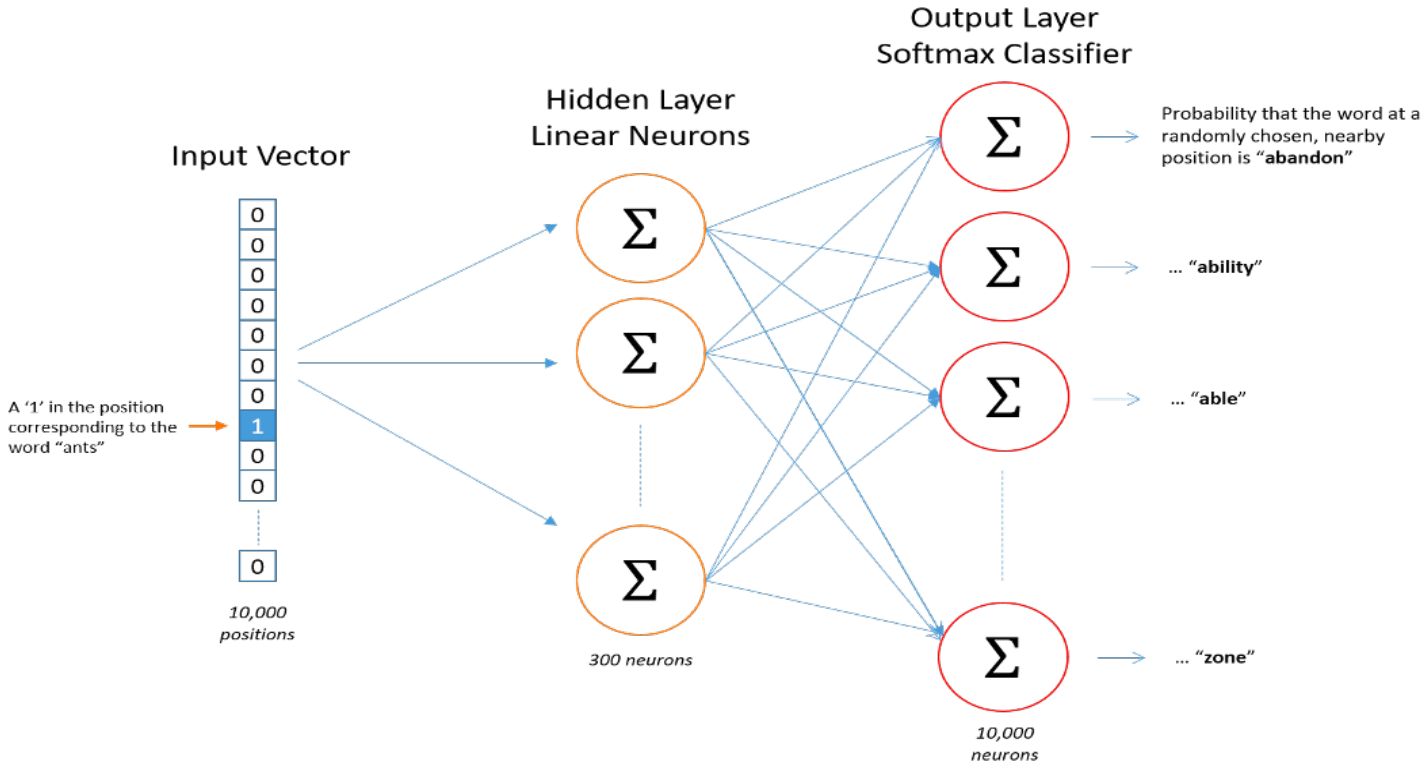
7 people found this helpful

Image source: https://thegradient.pub/understanding-evaluation-metrics-for-language-models/

Amazon comments for Uno Playing Cards

word2vec

- Shallow Neural Network architecture.
- Input: Text Corpus
- Output: w_i



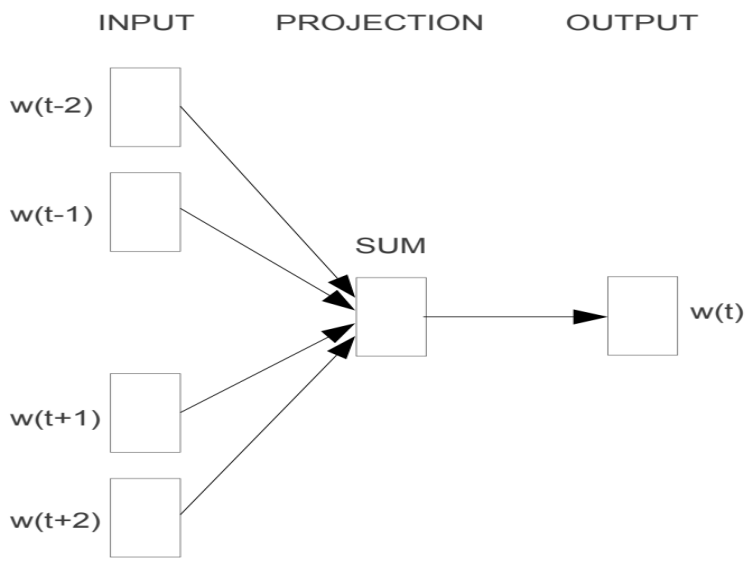
Softmax Function

$$\sigma(x_j) = \frac{e^{x_j}}{\sum_i e^{x_i}}$$

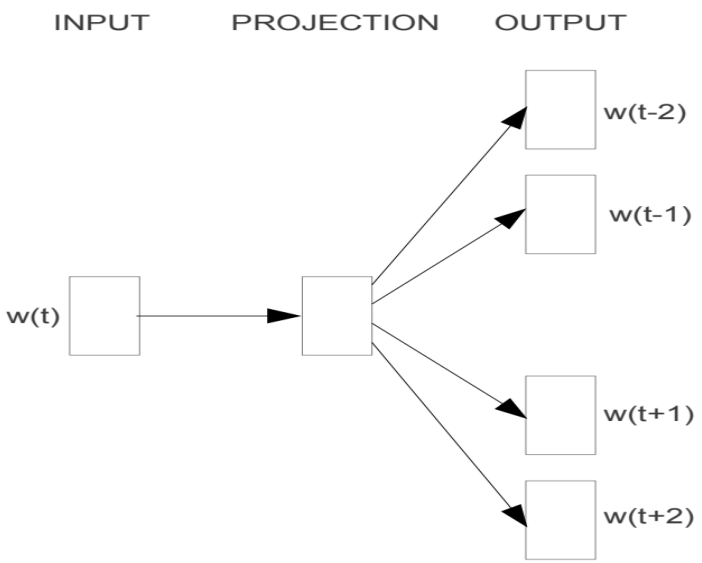
Image source: https://towardsdatascience.com/word2vec-skip-gram-model-part-1-intuition-78614e4d6e0b

word2vec

- How are they constructed?
  - **Continuous Bag of Words** (CBOW): Predict current word based on context
  - **Skip-gram**: Predict surrounding words from a given word



CBOW



Skip-gram

Image source: Efficient Estimation of Word Representations in Vector Space", 2013

https://www.tensorflow.org/tutorials/text/word_embeddings

Word Embeddings Tutorial

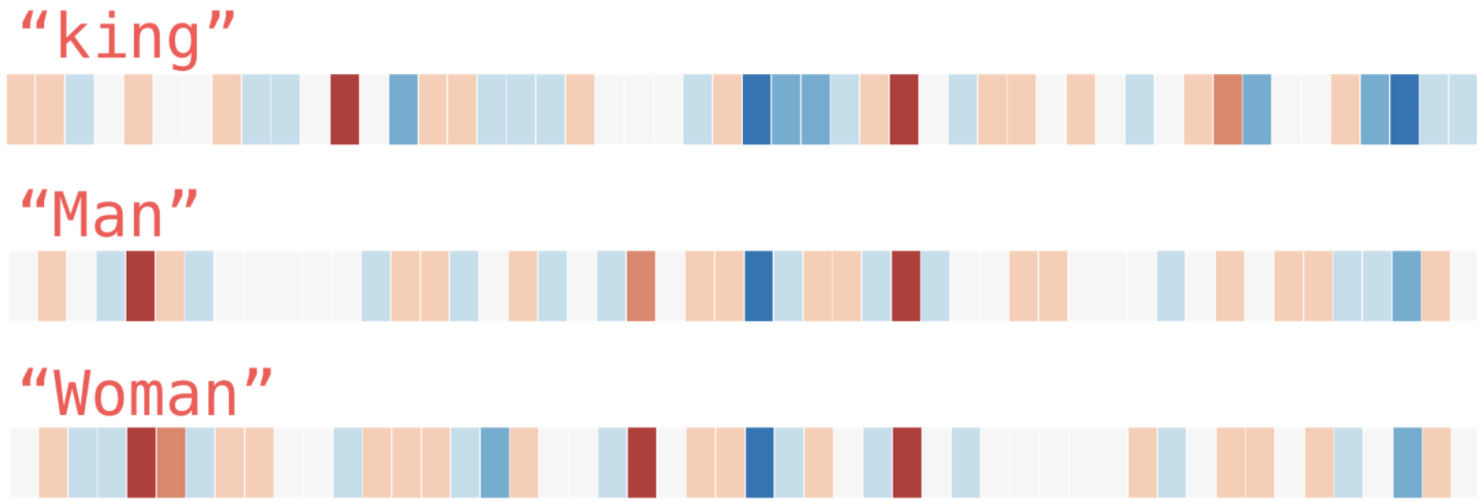
GloVe

- Derive the relationship between the words from global statistics
  - Difference from word2vec: Look at co-occurrence statistics explicitly!
  - No need of neural network
- Co-occurrence matrix
- Encode the information of the probability ratio - probability that word *j* appear in the context of word *i* with respect to probe words

Probability and Ratio	$k = solid$	$k = gas$	$k = water$	$k = fashion$
$P(k\|ice)$	1.9×10^{-4}	6.6×10^{-5}	3.0×10^{-3}	1.7×10^{-5}
$P(k\|steam)$	2.2×10^{-5}	7.8×10^{-4}	2.2×10^{-3}	1.8×10^{-5}
$P(k\|ice)/P(k\|steam)$	8.9	8.5×10^{-2}	1.36	0.96

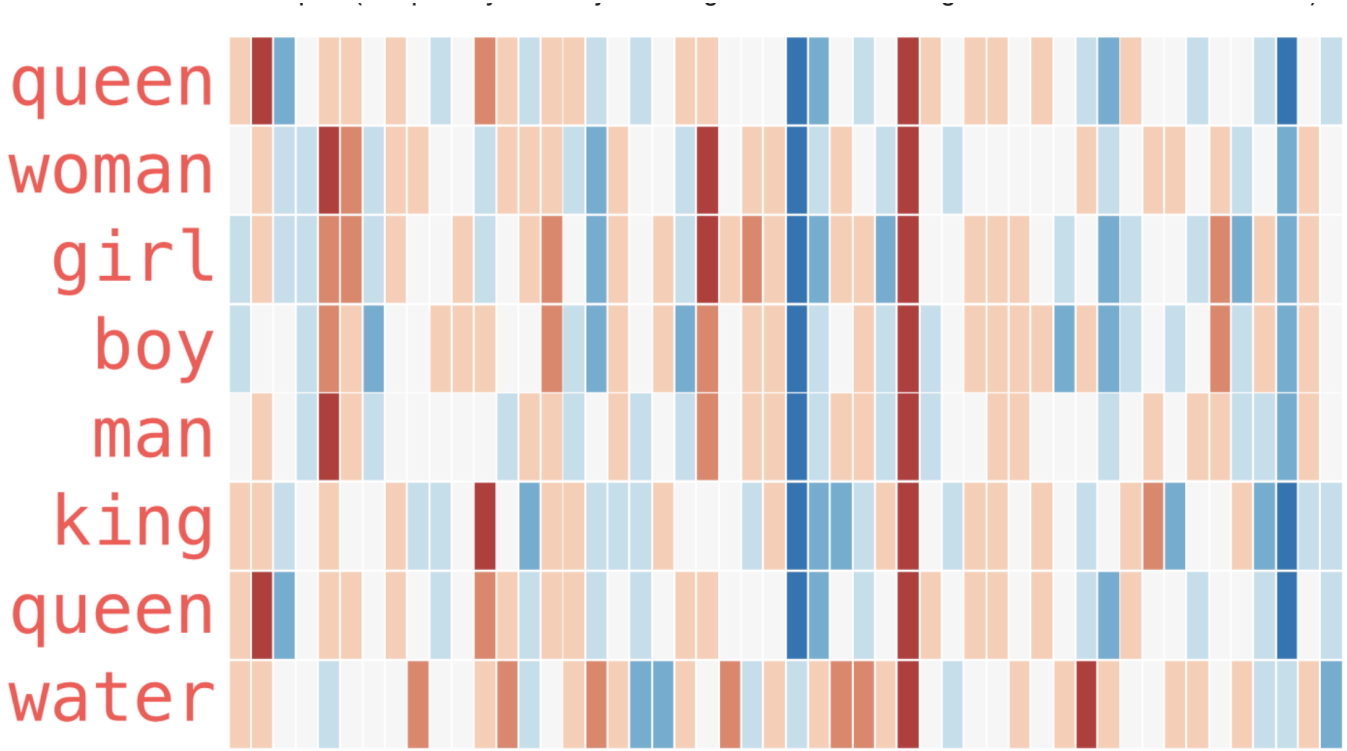
Paper: https://nlp.stanford.edu/pubs/glove.pdf

GloVe



http://jalammar.github.io/illustrated-word2vec/

GloVe



http://jalammar.github.io/illustrated-word2vec/

ELMo

- Contextual:
  - Representation of a word is no longer constant!
  - Depends on the context in which the word appears
  - *River bank* vs *financial bank*
- Character-based
  - Lesser hassle with OOV words
- Model: Bidirectional LSTM
- Representations: Function of all the internal layers of the biLM

Paper: https://arxiv.org/abs/1802.05365

Using ELMo for supervised tasks

Tutorial!

Using Word Embeddings

Learn from corpus, then use it as input

Learn the embedding layer jointly with the main model task

Use pretrained embeddings

Seed with pretrained embeddings, then train with the model

Questions?

Pre-processing: Byte-Pair Encoding

What? Replace common pairs of consecutive bytes with a byte that does not appear in that data.

Why? Rare words' representation

How? Start from character level and build up using frequency of occurrence till desired vocab size is reached

Paper: Sennrich, R., Haddow, B., & Birch, A. (2015). Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*.

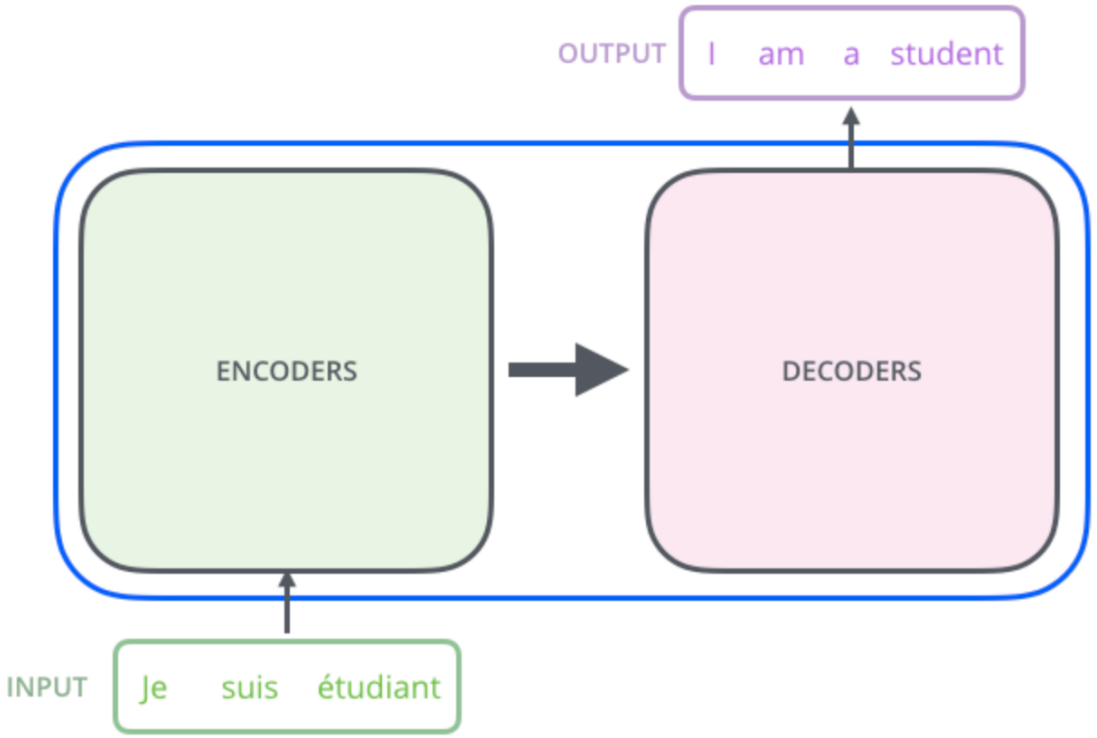
Pre-processing: WordPiece

What? Replace common pairs of consecutive bytes with a byte that does not appear in that data.

Why? Rare words' representation

How? Start from character level and build up using likelihood of the possible candidates till desired vocab size is reached

Paper: Sennrich, R., Haddow, B., & Birch, A. (2015). Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*.



Seq2seq models

Transformer

Introduced in Attention is All You Need by Vasvani et al.*

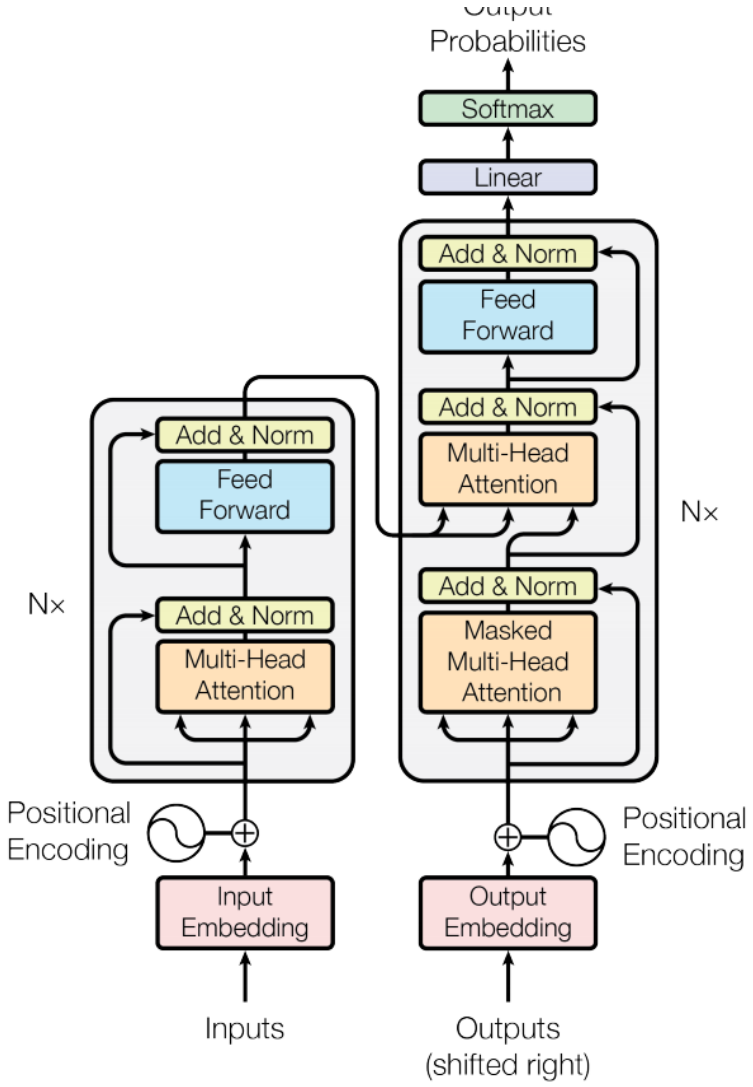
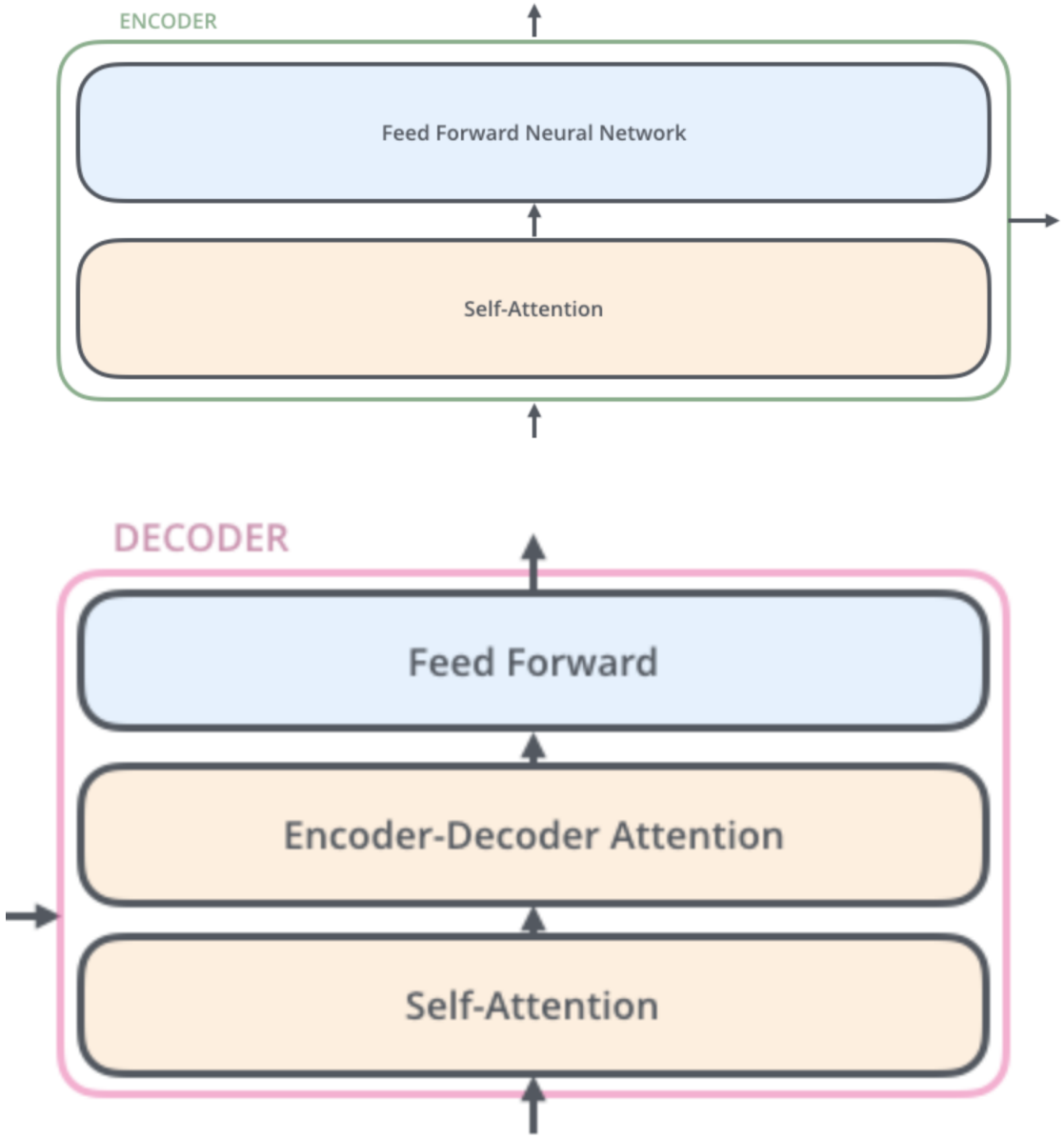


Figure 1: The Transformer - model architecture.

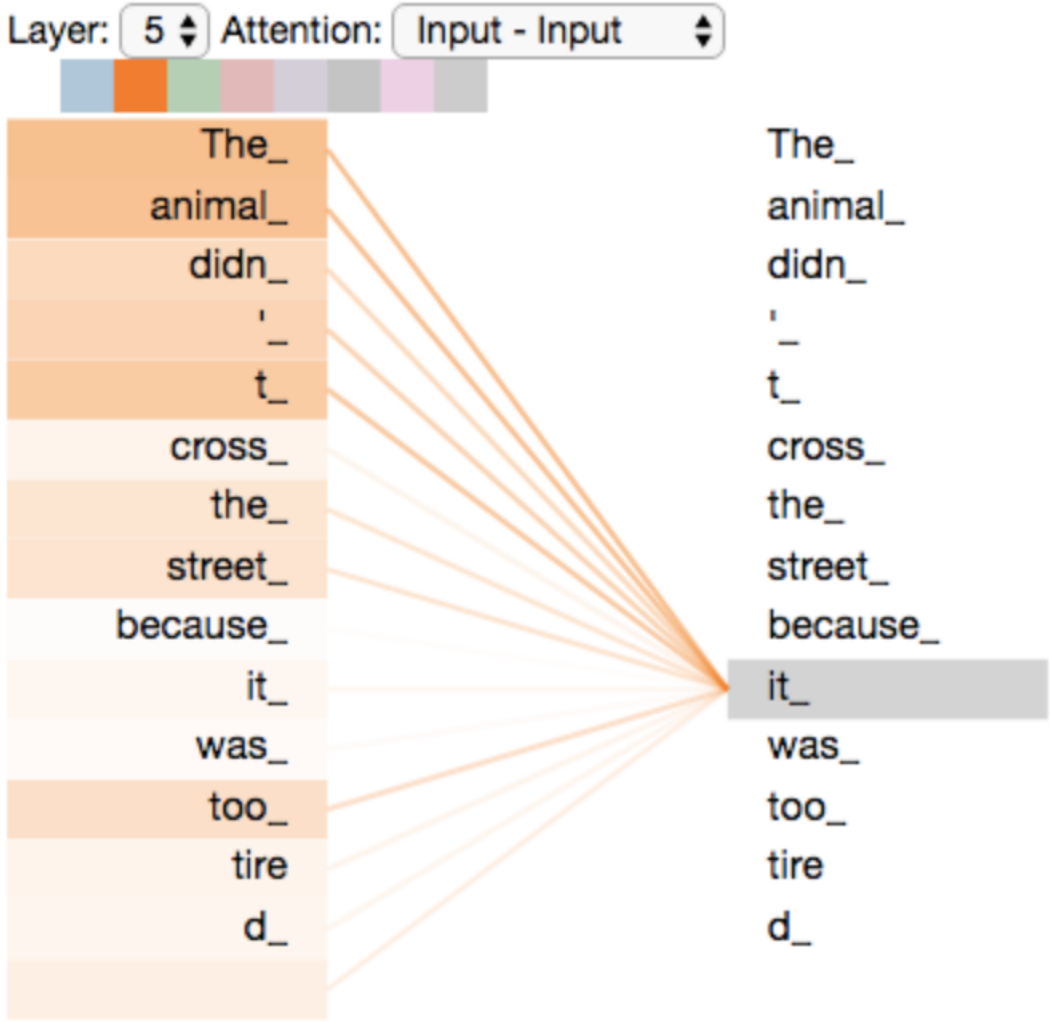
https://arxiv.org/abs/1706.03762

Transformer



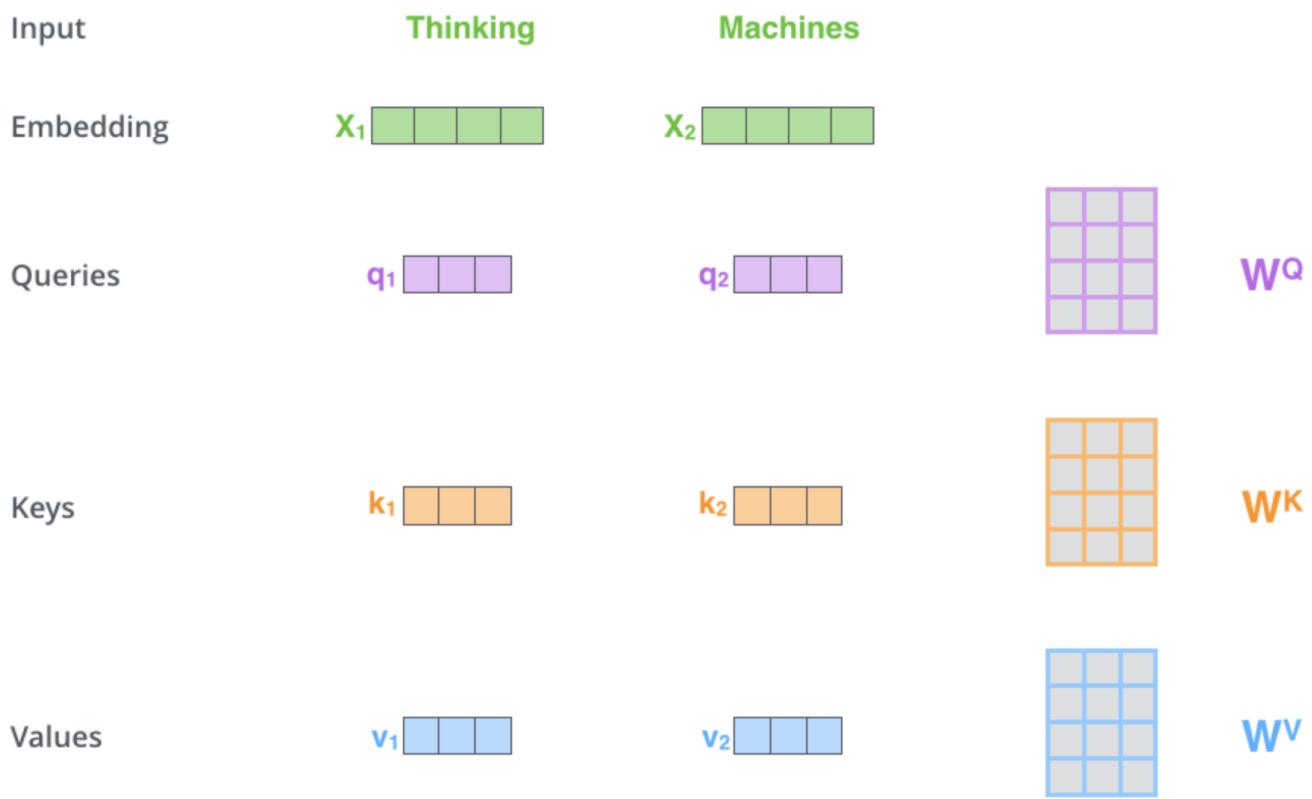
http://jalammar.github.io/illustrated-transformer/

Attention



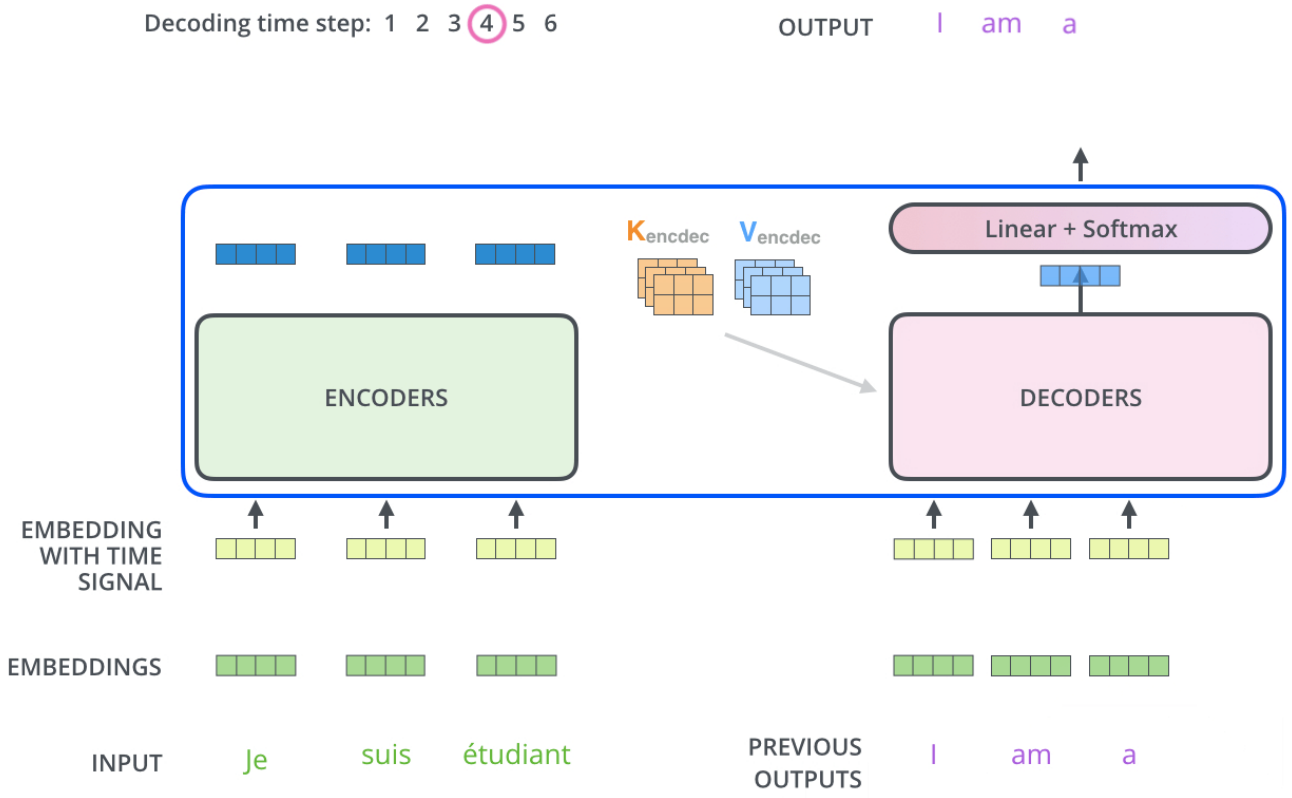
http://jalammar.github.io/illustrated-transformer/

Attention



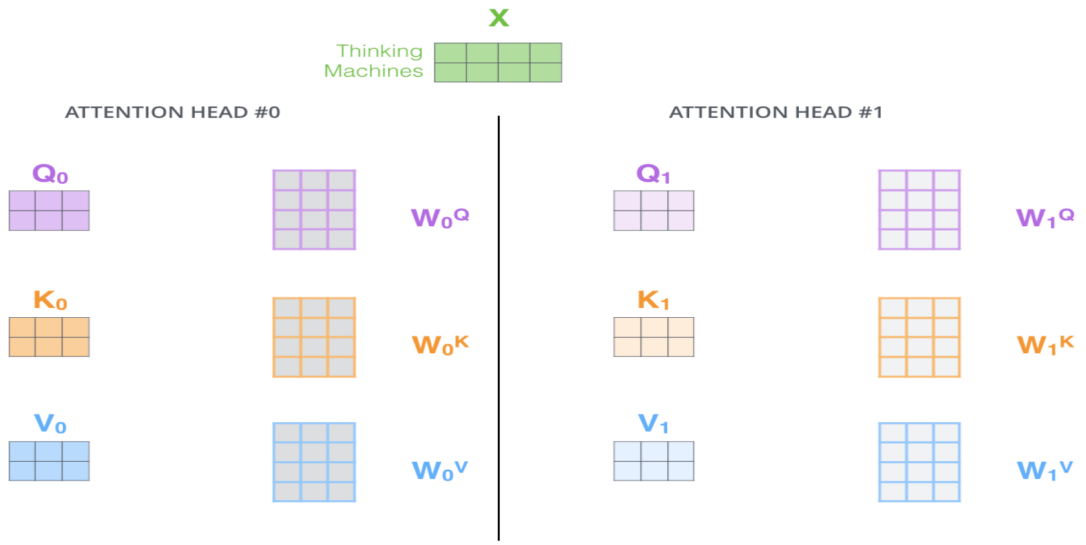
http://jalammar.github.io/illustrated-transformer/

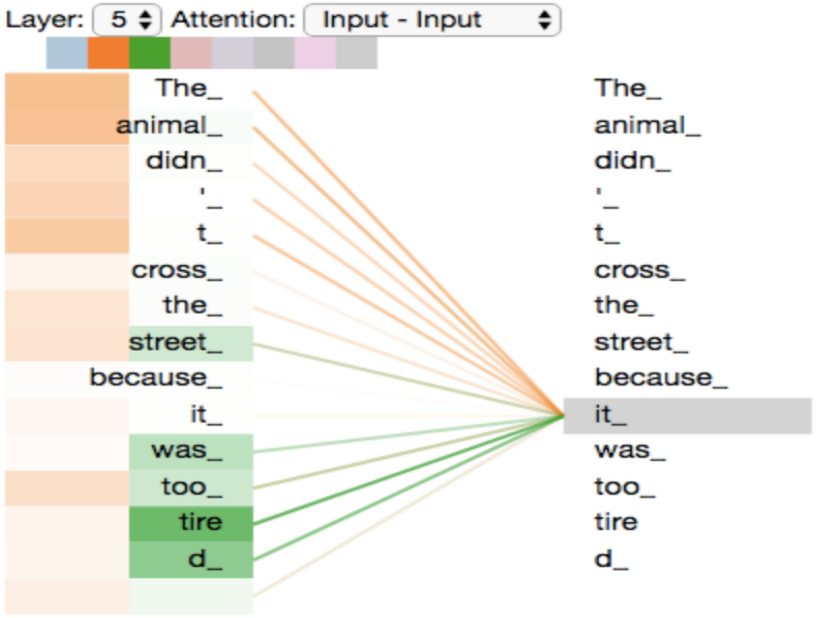
Attention



http://jalammar.github.io/illustrated-transformer/

Multi-headed Attention



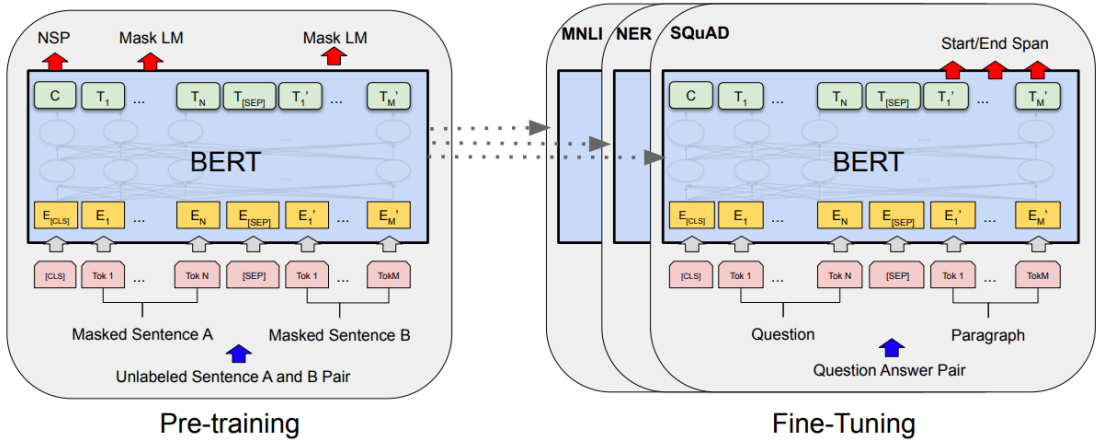


http://jalammar.github.io/illustrated-transformer/

Questions?

BERT - Tasks

- Pre-training: Train on unlabeled data
  - Next sentence prediction:
    - True label: IsNext 50% of the time
    - False Label: as NotNext
  - Masked LM
- Fine-tuning: Initialize with pre-trained weights and fine-tune for task with labeled data
  - Sentence pairs in paraphrasing
  - Hypothesis-premise pairs
  - Question-passage pairs in question answering
  - A degenerate text-Ø pair in text classification or sequence tagging.



BERT – Pre-Training

- Input
  - A single sentence
  - Pair of sentences (e.g.,< Question, Answer>): Separated by <SEP> and a learned embedding is assigned to every token indicating whether it belongs to sentence A or sentence B
  - 30k vocab
  - WordPiece Embedding
- Data: BooksCorpus (800M words) (Zhu et al., 2015) and English Wikipedia (2,500M words)

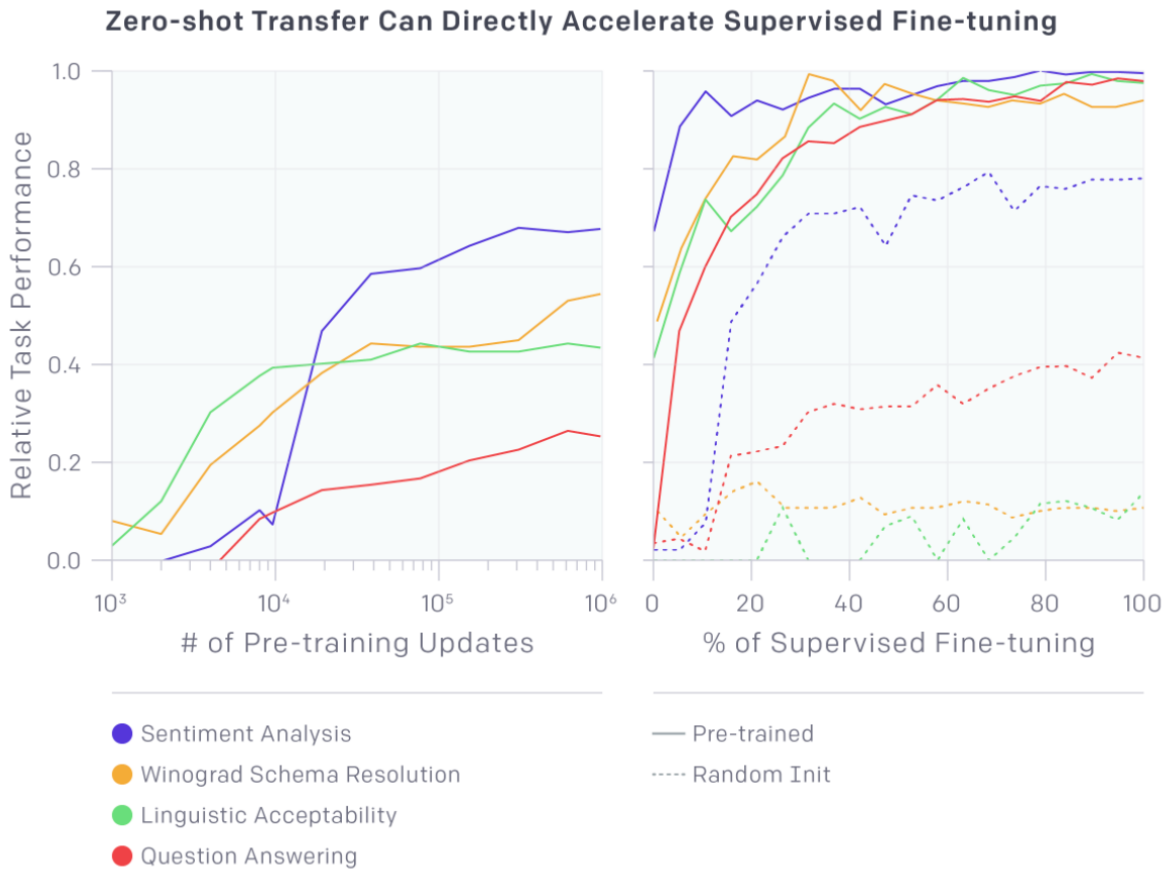
BERT – Masked Language Model

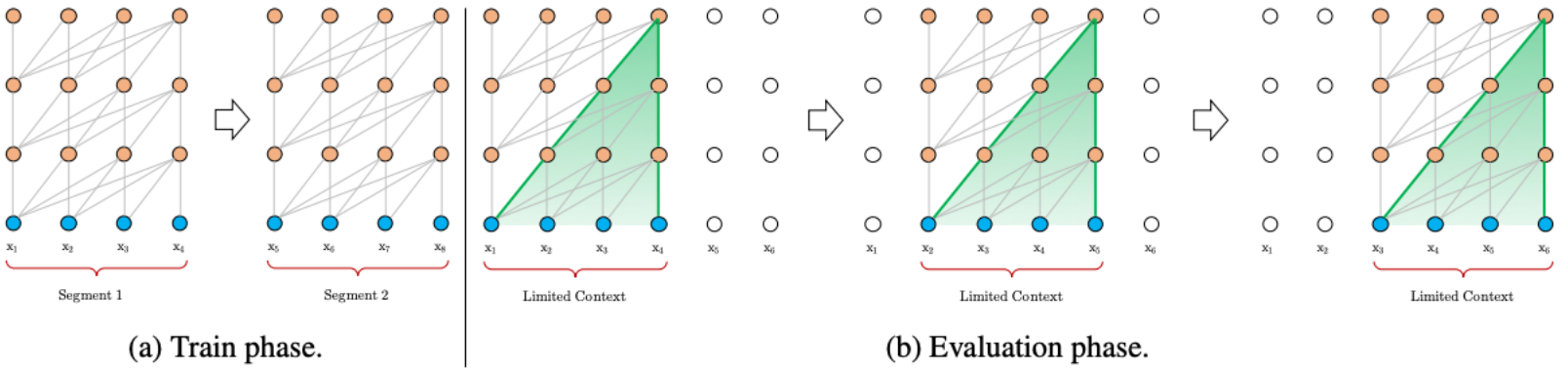
- Bidirectional
- Mask 15% of all WordPiece tokens in each sequence at random. Replace masked token with :
  - 80% probability by: [MASK]
  - 10% probability by: <random token>
  - 10% probability by: <original token>
- Evaluate by predicting the masked words

Other Models

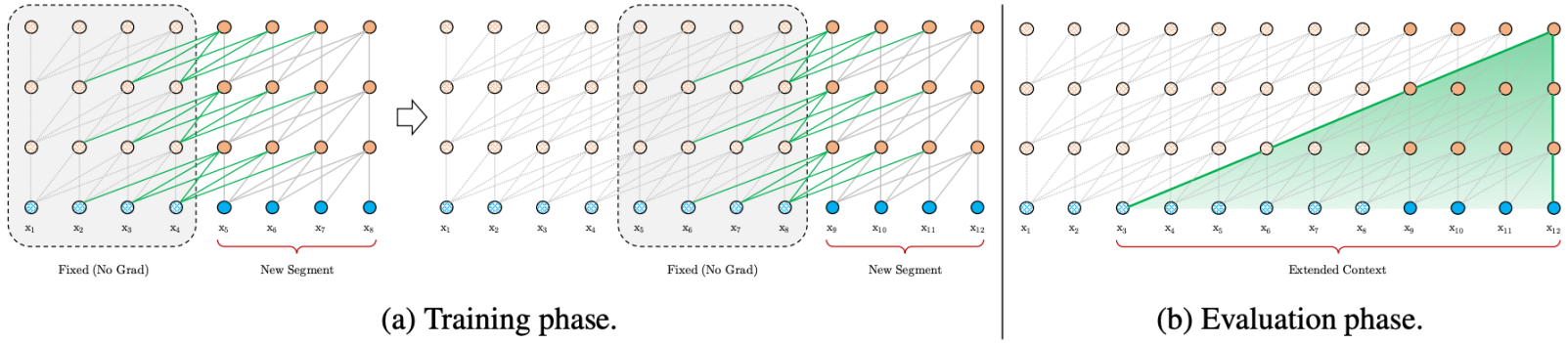
- GPT (OpenAI)
  - Train transformer for unsupervised tasks
  - Then fine-tune on supervised datasets
  - Pre-training step - 1 month on 8 GPUs

- Transformer-XL (Google/CMU)
  - Learn dependency beyond a fixed length without disrupting temporal coherence
  - Learns dependency that is 80% longer than RNNs and 450% longer than vanilla Transformer
  - Achieves better performance on both short and long sequences
  - Upto 1,800+ times faster than vanilla Transformers during evaluation.





Vanilla Transformer

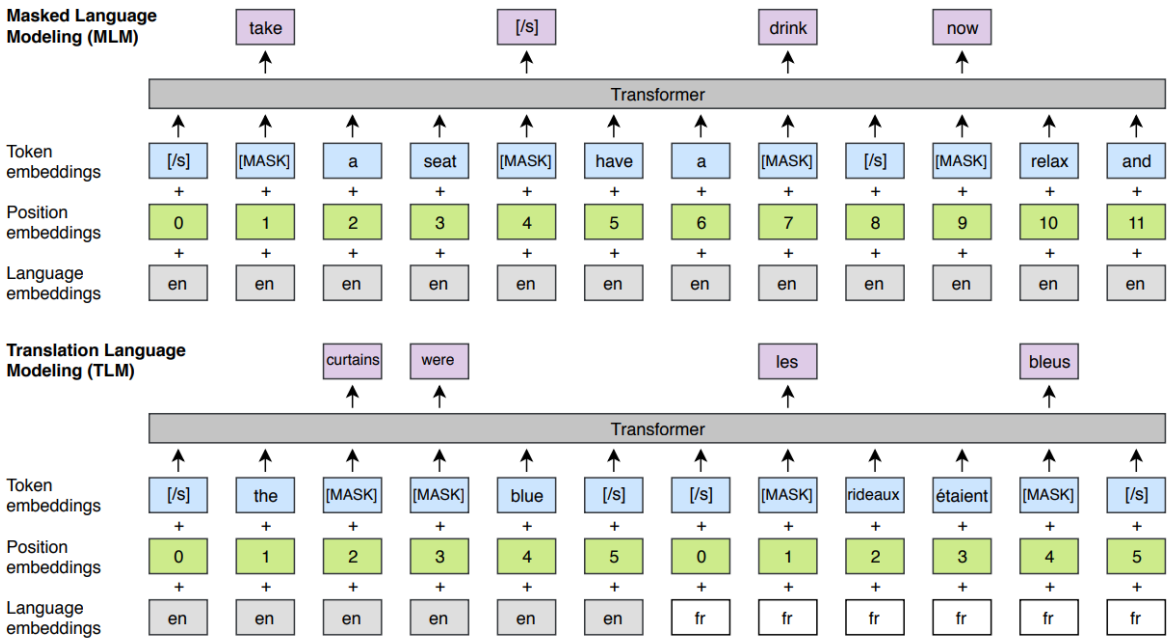


Transformer XL

Other Models: Transformer XL

Other Models: XLM

- Cross-lingual language model pretraining (XLM)
- Training one model for many languages while not sacrificing per-language performance
- Propose two methods
  - Unsupervised learning on monolingual data
  - Supervised language on parallel data



Key Takeaways

- What are word embeddings?
- Which popular pre-trained embeddings exist?
- How to create your own embeddings?
- What is a Transformer?
- How do we get BERT representations?
- What techniques are used by these state-of-the-art models?

Key Takeaways

You do not have to re-invent the wheel!

References

- Tensorflow Projector
- Word2vec paper: https://arxiv.org/pdf/1301.3781.pdf
- GloVe code and embeddings: https://nlp.stanford.edu/projects/glove/
- https://tfhub.dev/google/elmo/3
- https://github.com/allenai/allennlp
- Intuition behind Transformer: http://jalammar.github.io/illustrated-transformer/

WOMEN WHO CODE PYTHON

Thank you!

Shreya Khurana

Data Scientist at GoDaddy

https://github.com/ShreyaKhurana