UVA CS 4774: Machine Learning

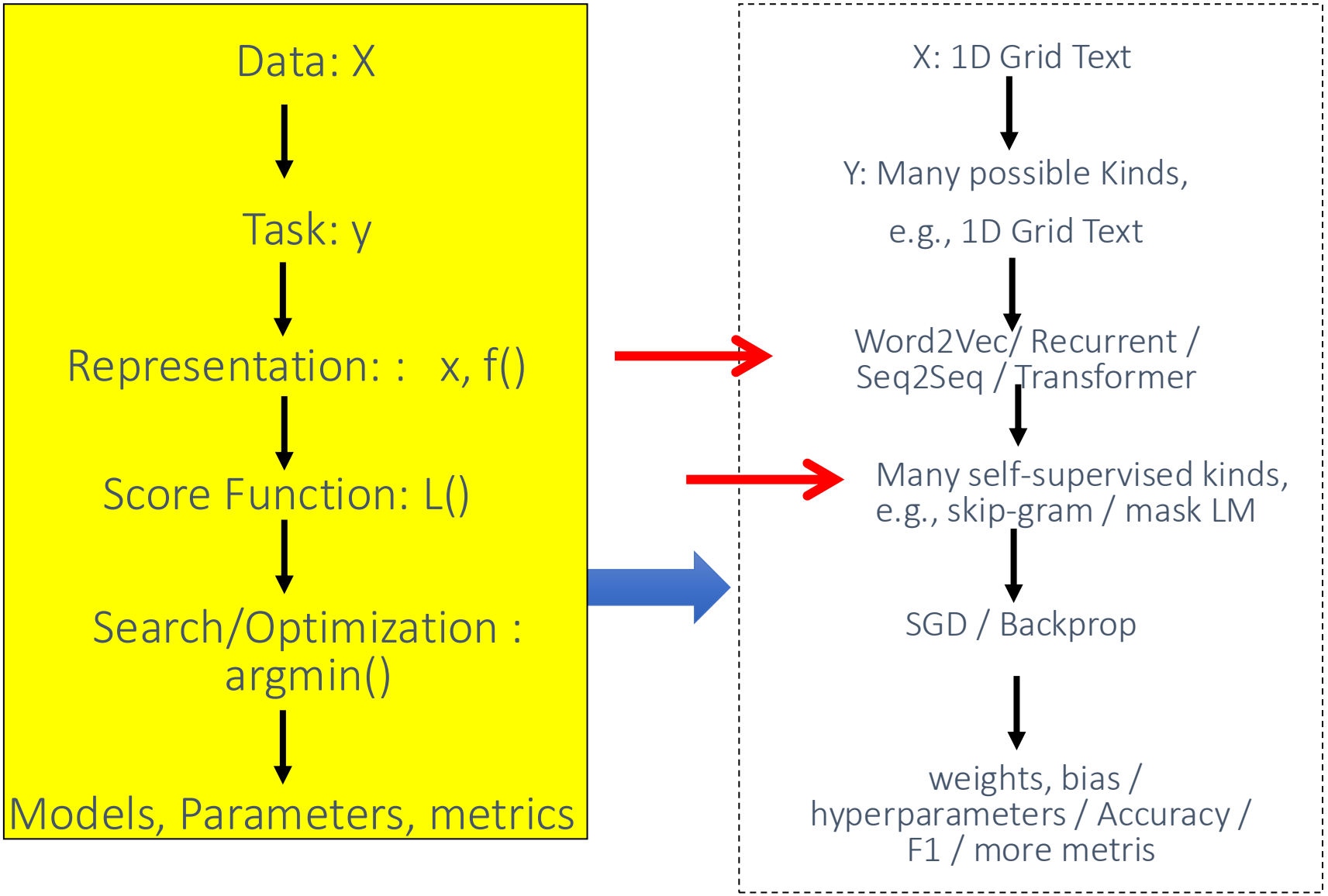
S3: Lecture 17: Deep Seq2Seq Networks for Natural Language Processing

Dr. Yanjun Qi

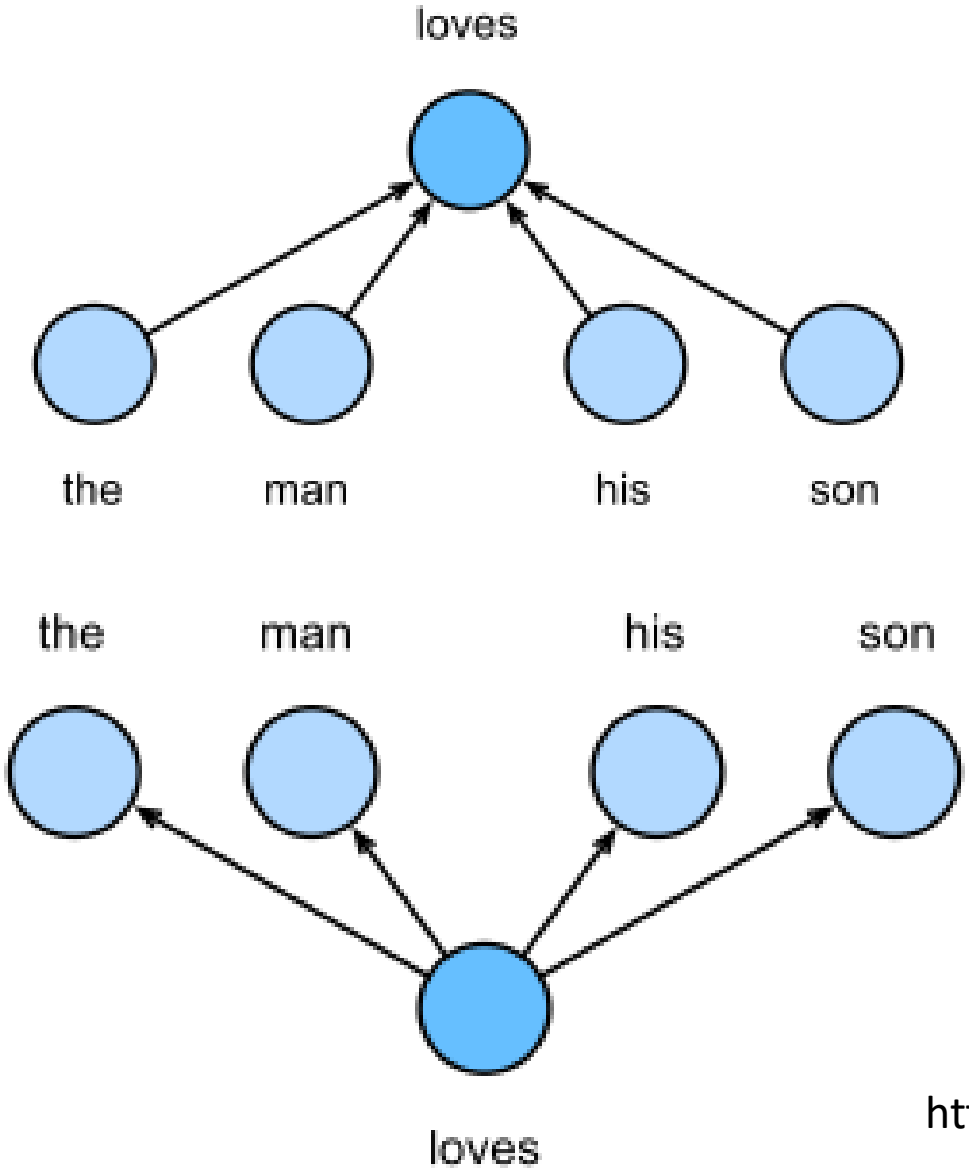
University of Virginia
Department of Computer Science

Module I

Today: Neural Network Models on 1D Grid / Language Data



Recap of Word2Vec:



Essentially, skip-gram and CBOW share the same two weight tables. They only differ in the objective function during training.

https://en.wikipedia.org/wiki/Word2Vec

Training Signals: the vector of a word should be close to the vector of each of its neighbors.

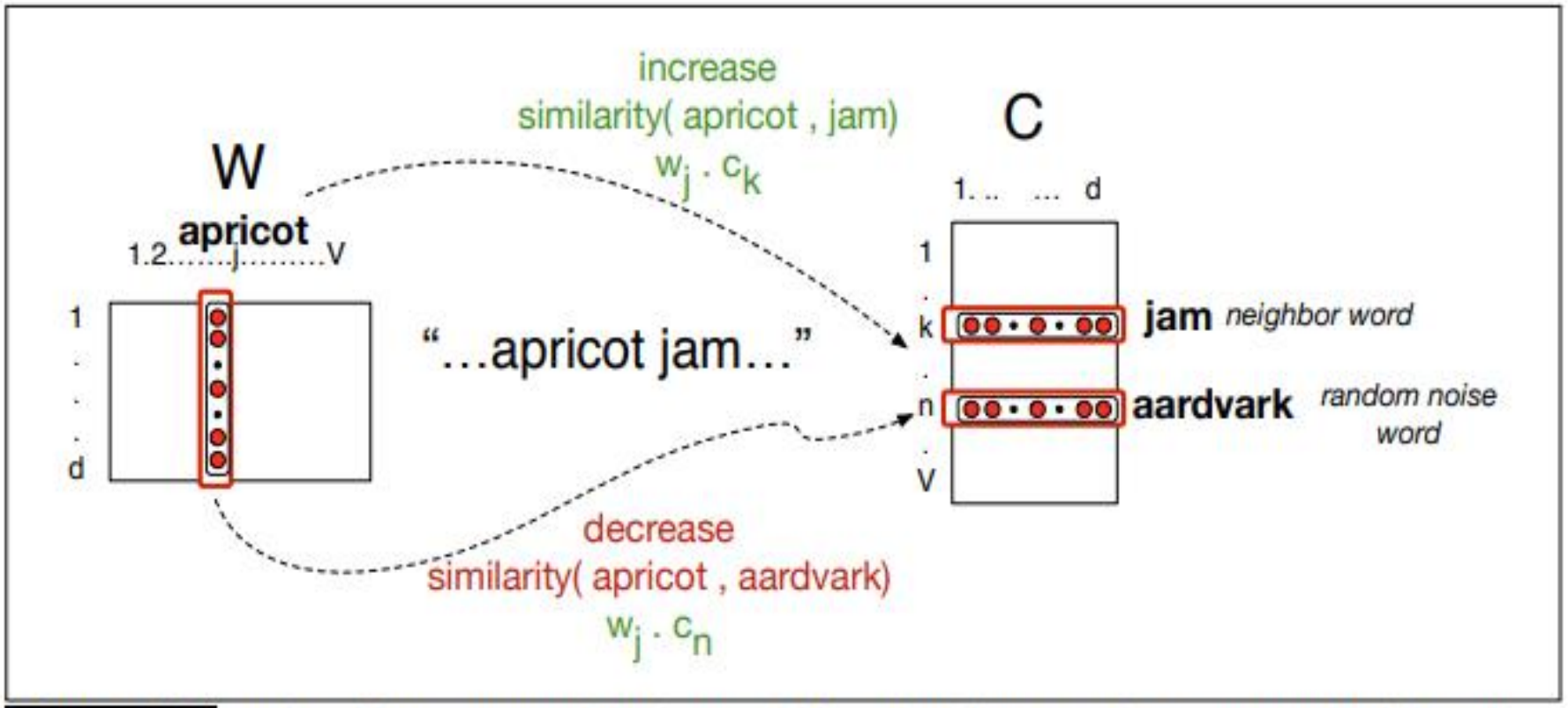


Figure 6.13 The skip-gram model tries to shift embeddings so the target embedding (here for *apricot*) are closer to (have a higher dot product with) context embeddings for nearby words (here *jam*) and further from (have a lower dot product with) context embeddings for words that don't occur nearby (here *aardvark*).

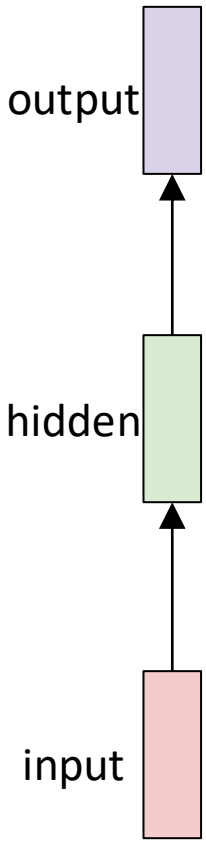
Negative Sampling based Binary Cross-Entropy Loss (or Log-Sigmoid Loss) framed as a binary classification task.

$$J_{\text{NEG}} = -\log \sigma(u_{w_{\text{context}}}^{\top} v_{w_{\text{target}}}) - \sum_{i=1}^{K} \log \sigma(-u_{w_{n_i}}^{\top} v_{w_{\text{target}}})$$

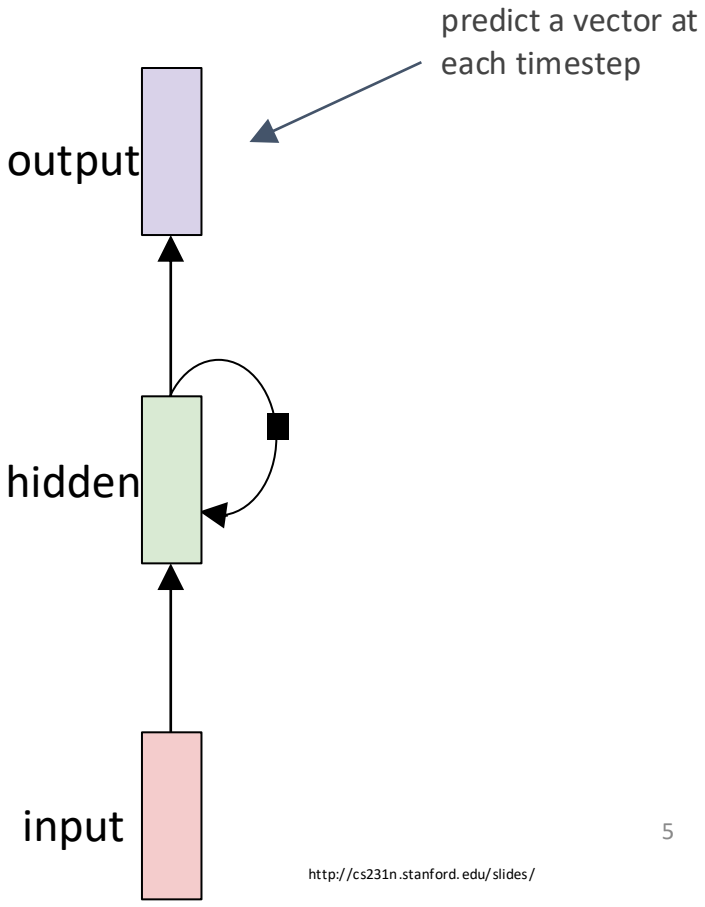
Recap of RNN models dynamic temporal dependency

- Make fully-connected layer model each unit recurrently
- Units form a directed chain graph along a sequence
- Each unit uses recent history and current input in modeling

Traditional "Feed Forward" Neural Network



Recurrent Neural Network



http://cs231n.stanford.edu/slides/

Roadmap : f() on natural language inputs

- Before Deep NLP (Pre 2012)
  - (BOW / LSI / Topic LDA)
- Word2Vec (2013-2016)
  - (GloVe/ FastText)
- Recurrent NN (2014-2016)
  - LSTM
  - Seq2Seq
- Attention / Self-Attention (2014 – now)
  - Attention
  - Transformer (self-attention, attention only)
  - BERT / XLNet/ GPT-2 / T5 …

Variable Length Nature of Natural Language

This Food is not good.



This wonderful book is a pleasure to read.



Variable-length input / output examples:

Anything requiring long-range patterns	Anything generative
• Question detection • Natural language context understanding • Entity disambiguation • Sentence embedding	• Machine translation • Natural language generation • Question answering • Skip-thoughts

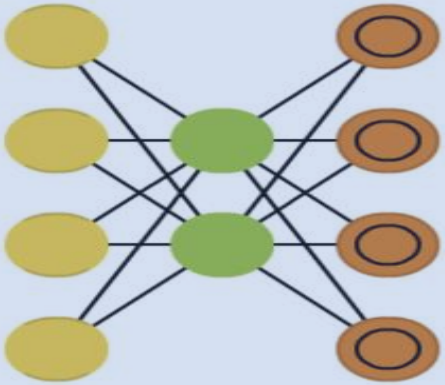
http://cs231n.stanford.edu/slides/

Encoder

Decoder

$\vec{x}$ → $\vec{h}$ → $\vec{\hat{x}}$

Auto Encoder (AE)



Minimize diff

$|\vec{\hat{x}} - \vec{x}|$

modeling variable-length input / output with

Seq2Seq

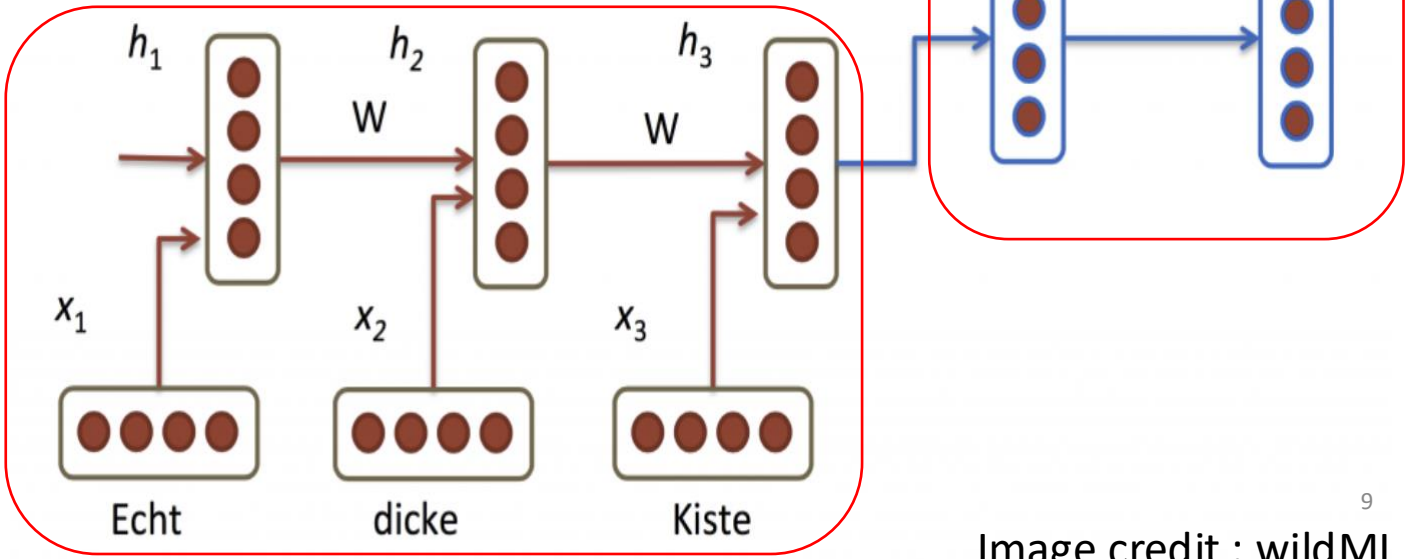


Image credit : wildML

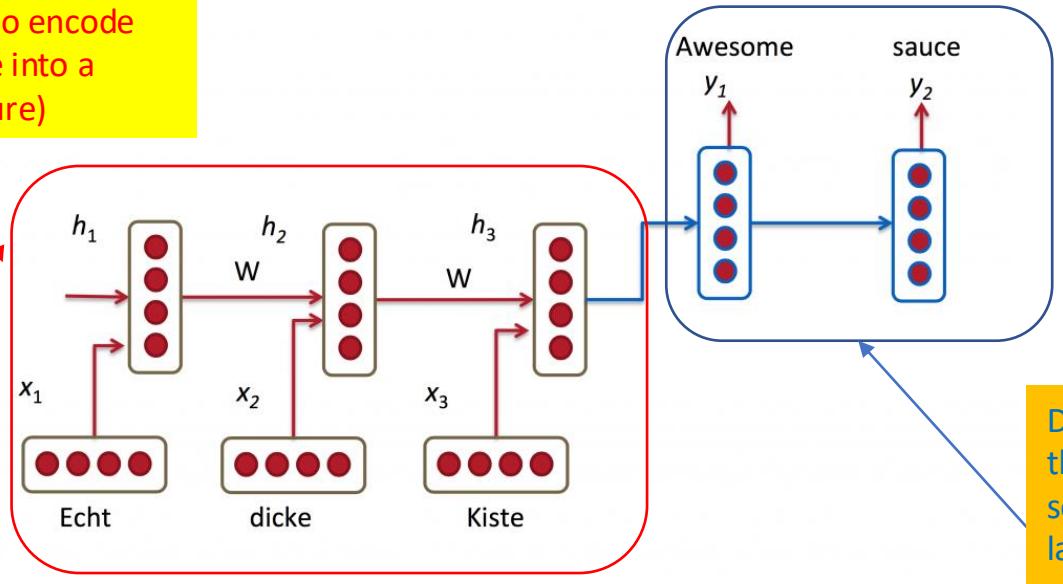
Seq2Seq for Machine Translation

In machine translation, the **input** is a sequence of words in source language, and the **output** is a sequence of words in target language.

For instance: Two RNN LSTMs for Machine Translation (German to English)

- Encoder LSTM (on German)
- Decoder LSTM (on English)

Encoder: An RNN to encode the input sentence into a hidden state (feature)

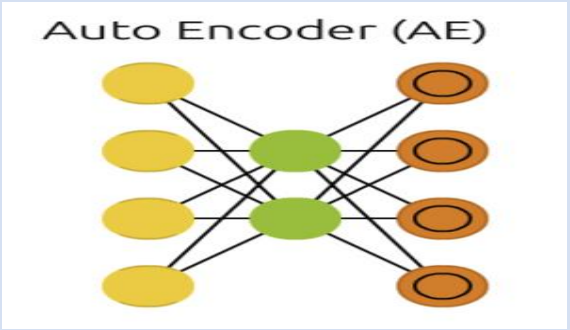


Decoder: An RNN with the hidden state of the sentence in source language as the input and output the translated sentence

Encoder-decoder architecture for machine translation

Adapt from Professor Qiang Yang of HK UST

$\vec{x}$ Encoder $\vec{h}$ Decoder $\vec{\hat{x}}$

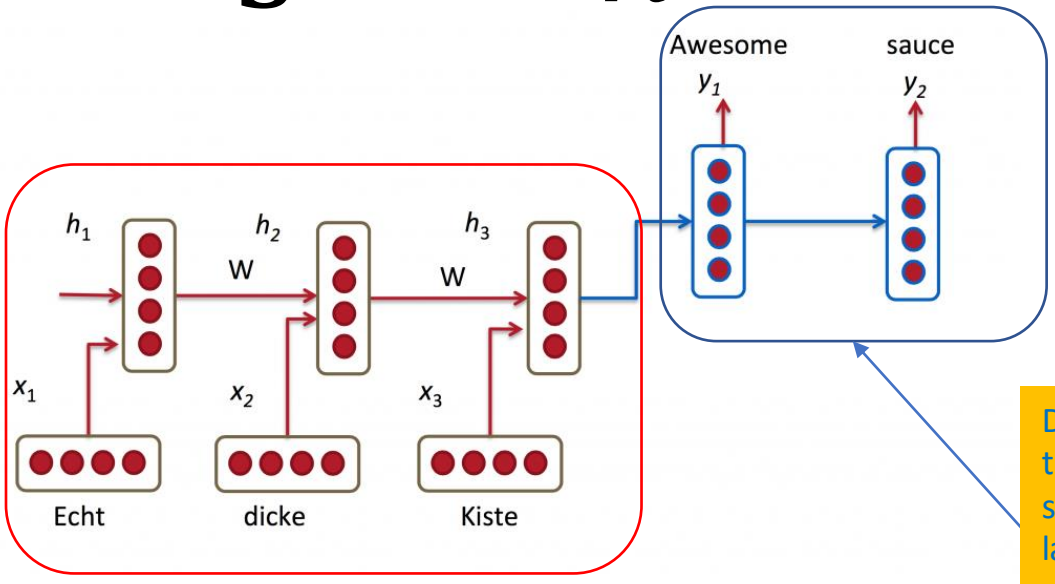


Minimize diff

$|\vec{\hat{x}} - \vec{x}|$

Encoder-decoder architecture for machine translation

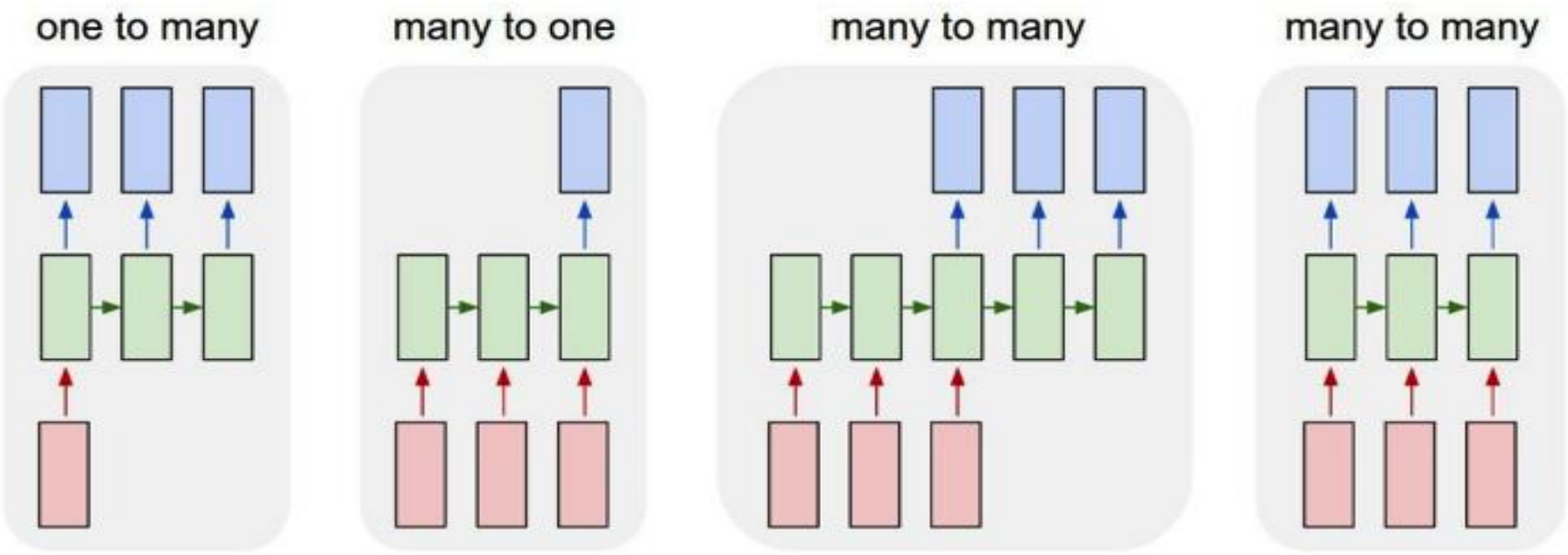
$\vec{s}$ Encoder $\vec{h}$ Decoder $\vec{\hat{t}}$



Encoder: An RNN to encode the input sentence into a hidden state (feature)

Decoder: An RNN with the hidden state of the sentence in source language as the input and output the translated sentence

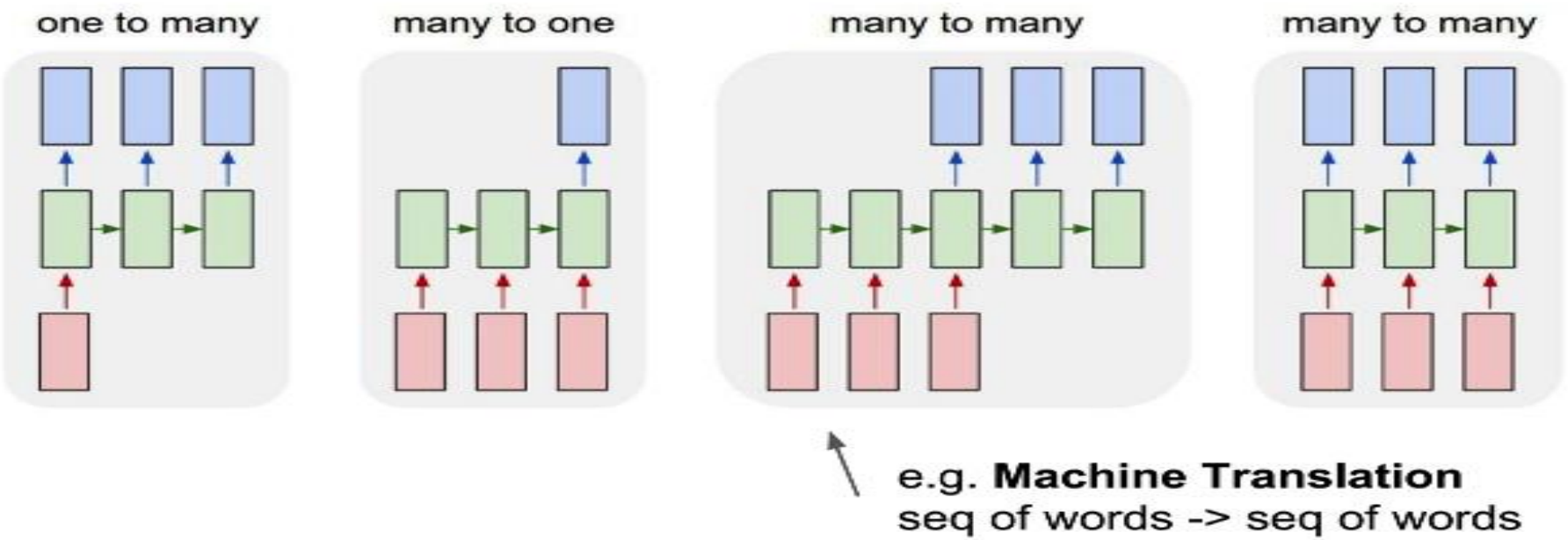
Seq2Seq (with RNNs) can handle

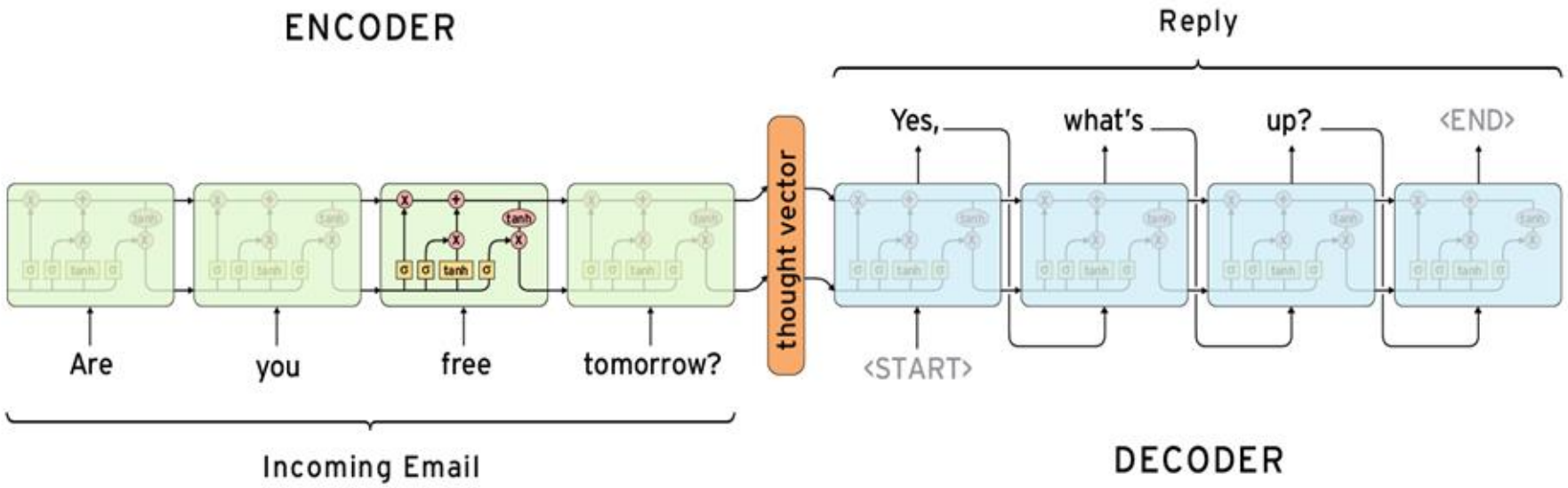


e.g. **Sentiment Classification**
sequence of words -> sentiment

http://cs231n.stanford.edu/slides/

Seq2Seq (RNNs) can handle





Seq2Seq can handle most NLP tasks:

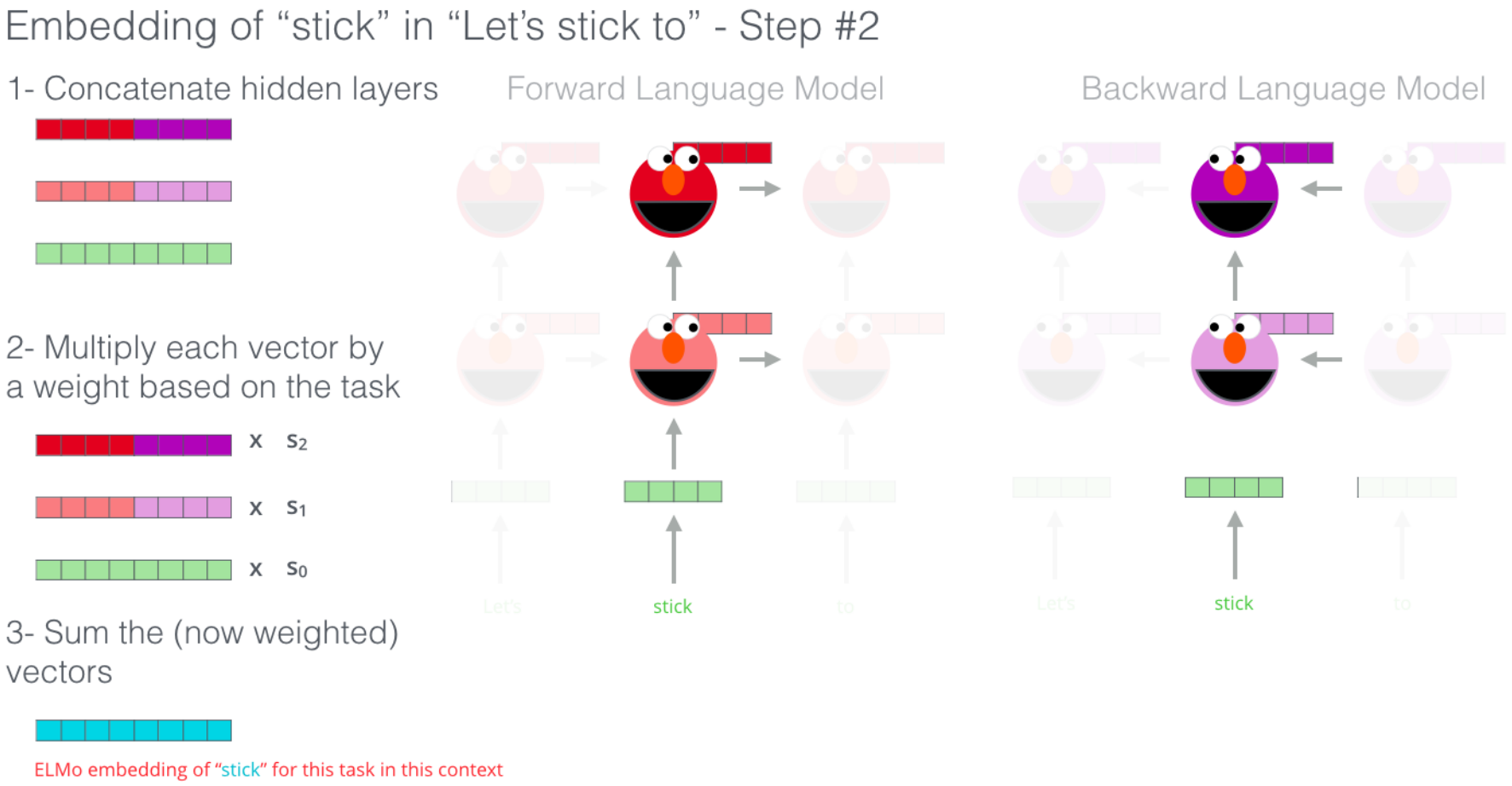
Given source sentences, learn an optimal model to automatically generate accurate and diversified target sentences that look like human generated sentences.



- Paraphrase generation: "How did xxx win the game?" → "How did xxx become the champion?"
- Dialogue generation: "You know French?" → "Sure do ... my Mom's from Canada"
- Question answering: "What was the name of the 1937 treaty?" → "Bald Eagle Protection Act"
- Style Transfer: "Just a dum funny question hahahaha" → "Just a senseless , funny question."

Adapt from Professor Qiang Yang of HK UST

Word contextual embedding with two RNNs' (bidirectional)



ELMo: Embeddings from Language Models

Pre-trained biLSTM for contextual embedding

UVA CS 4774: Machine Learning

S3: Lecture 17:

Deep Seq2Seq Networks for Natural Language Processing

Dr. Yanjun Qi

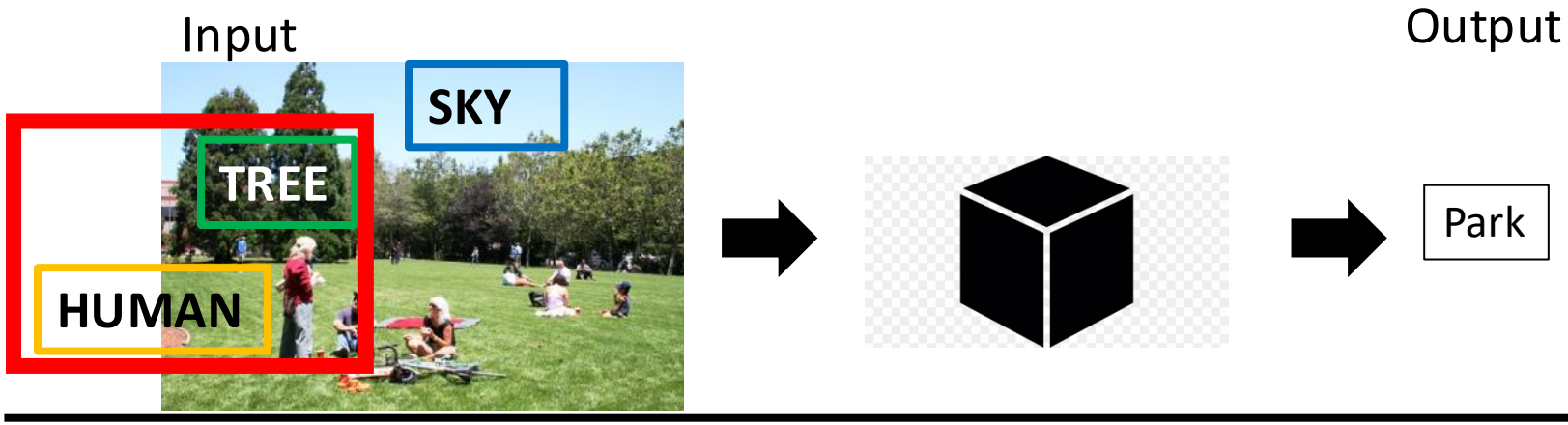
University of Virginia
Department of Computer Science

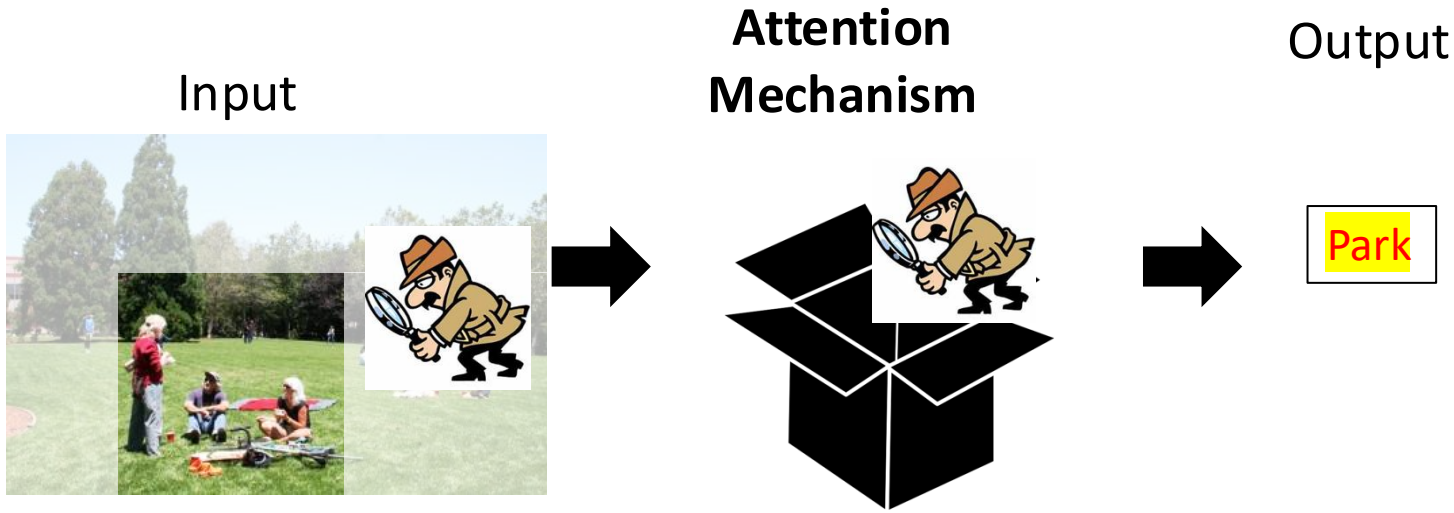
Module II

Roadmap : f() on natural language

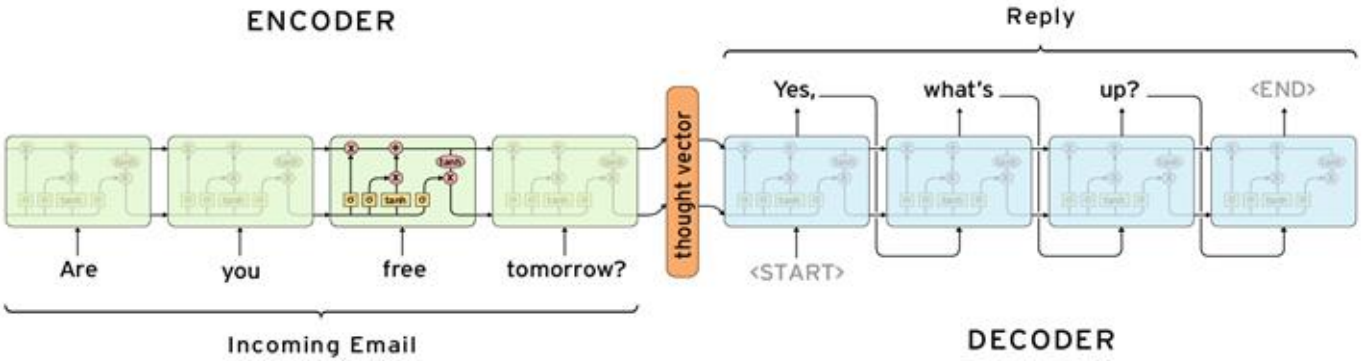
- Before Deep NLP (Pre 2012)
  - (BOW / LSI / Topic LDA)
- Word2Vec (2013-2016)
  - (GloVe/ FastText)
- Recurrent NN (2014-2016)
  - LSTM
  - Seq2Seq
- Attention / Self-Attention (2014 – now)
  - Attention
  - Transformer (self-attention, attention only)
  - BERT / XLNet/ GPT-2 / T5 …

Attention Trick

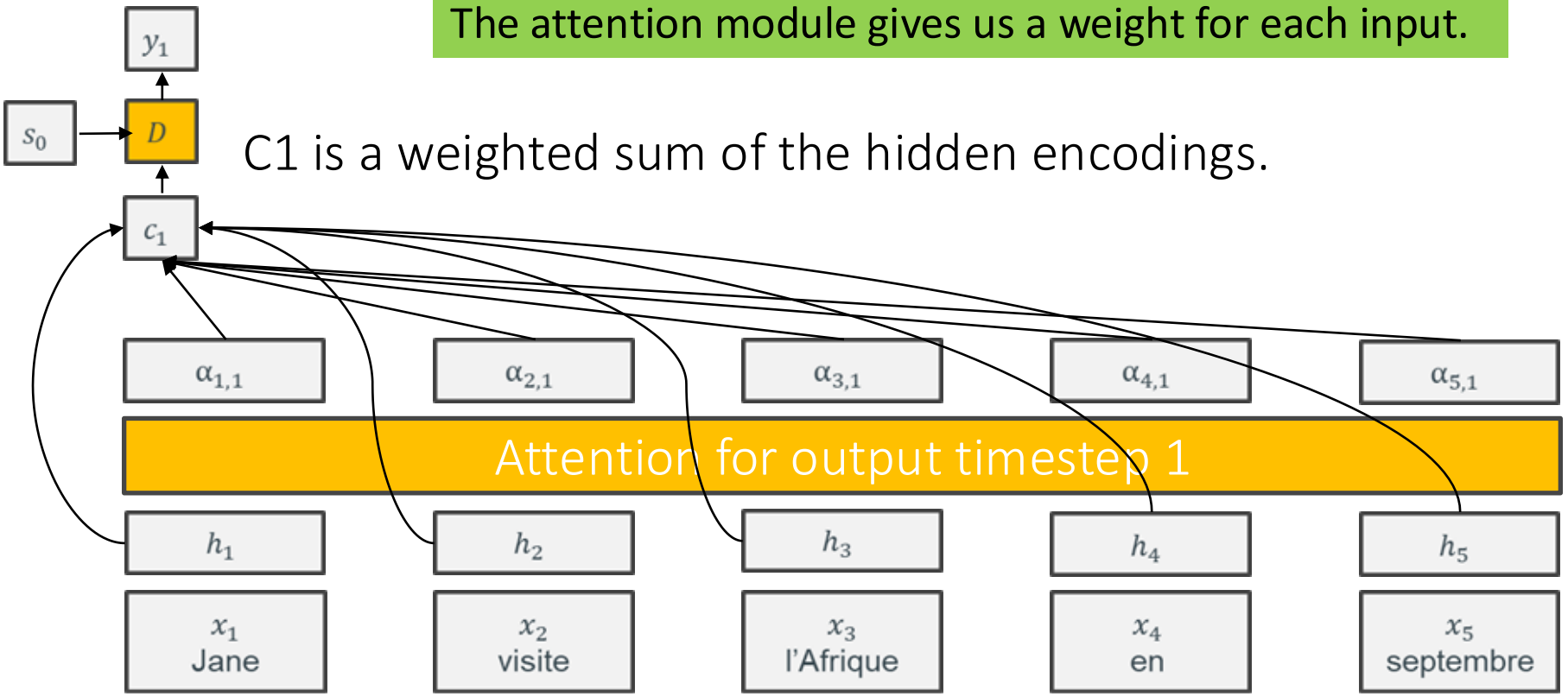




Attention Trick:

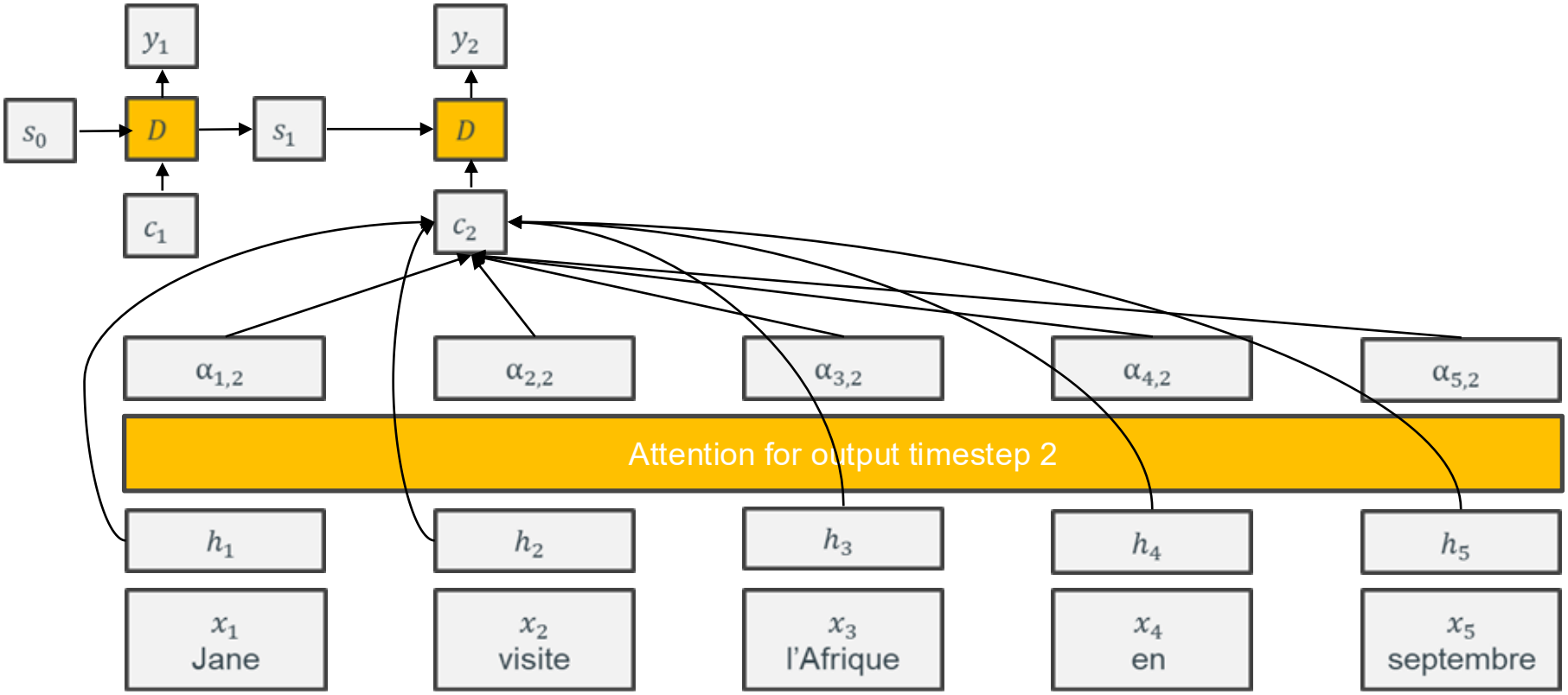


The attention module gives us a weight for each input.



Based: Dr. Yangqiu Song's slides

We then repeat for future timesteps.



Attention by hand: one output step

Decoder state s = [2, 1]. Score each input with a dot product: s · hi

Input word	**hi**	**Score s · hi**	**Weight αi,1 (softmax)**
Jane	[1, 0]	2	0.24
visite	[0, 1]	1	0.09
l'Afrique	[1, 1]	3	0.67

Context: ct = Σi αi,t hi

c1 = 0.24 [1, 0] + 0.09 [0, 1] + 0.67 [1, 1] = [0.91, 0.76]

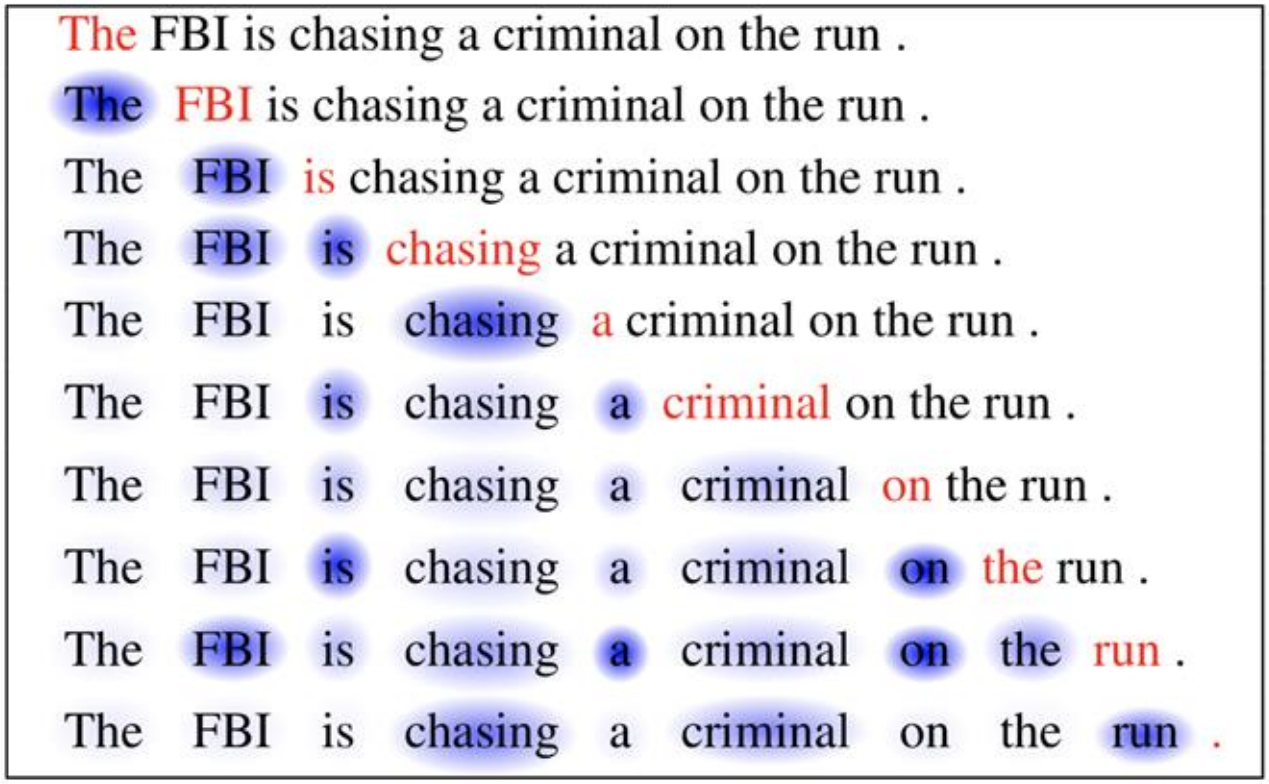
Weights are positive and sum to 1. The decoder gets a new c at every output step.

credit: worked example; numbers computed

Roadmap : f() on natural language

- Before Deep NLP (Pre 2012)
  - (BOW / LSI / Topic LDA)
- Word2Vec (2013-2016)
  - (GloVe/ FastText)
- Recurrent NN (2014-2016)
  - LSTM
  - Seq2Seq
- Attention / Self-Attention (2014 – now)
  - Attention
  - Transformer (self-attention, attention only)
  - BERT / XLNet/ GPT-2 / T5 …

Self-attention creates attention layers mapping from a sequence to itself.



Transformer: Exploiting Self Attentions

- A Google model (Vaswani et al. 2017).
  - Variable-length input
  - Variable-length output
  - **No recurrence**

- Uses 3 kinds of attention
  - Encoder self-attention.
  - Decoder self-attention.
  - Encoder-decoder multi-head attention.

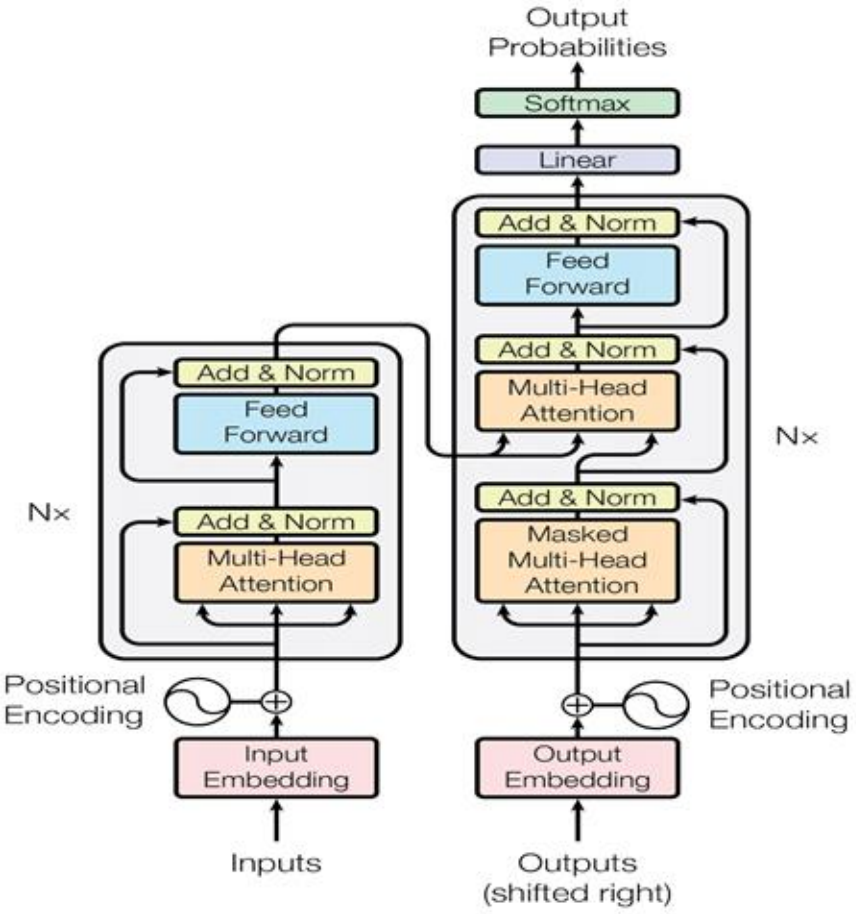


Figure 1: The Transformer - model architecture.

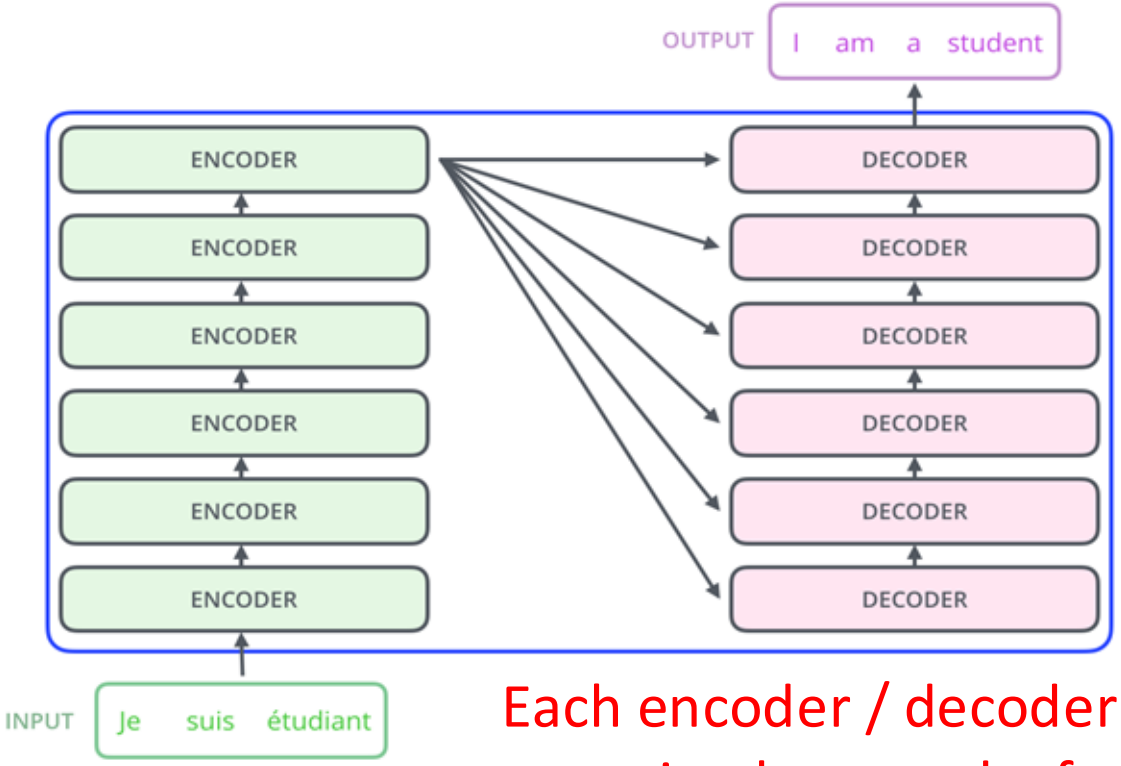
Three kinds of attention in one Transformer

Kind	Queries from	Keys, values from	Mask
Encoder self-attention	encoder	encoder	padding
Decoder self-attention	decoder	decoder	future + padding
Encoder-decoder attention	decoder	encoder	input padding

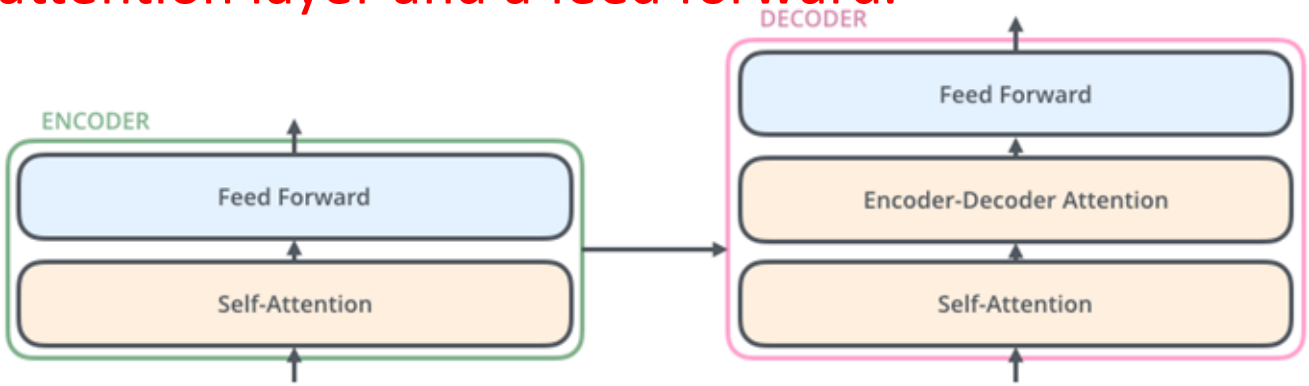
- Same formula each time: only the source of the vectors and the mask change
- Encoder-decoder attention is the attention trick from today, without an RNN
- BERT keeps the first row; GPT keeps the second

credit: Vaswani et al. 2017

Vanilla Transformer is Seq2Seq model



Each encoder / decoder layer has a self-attention layer and a feed forward.

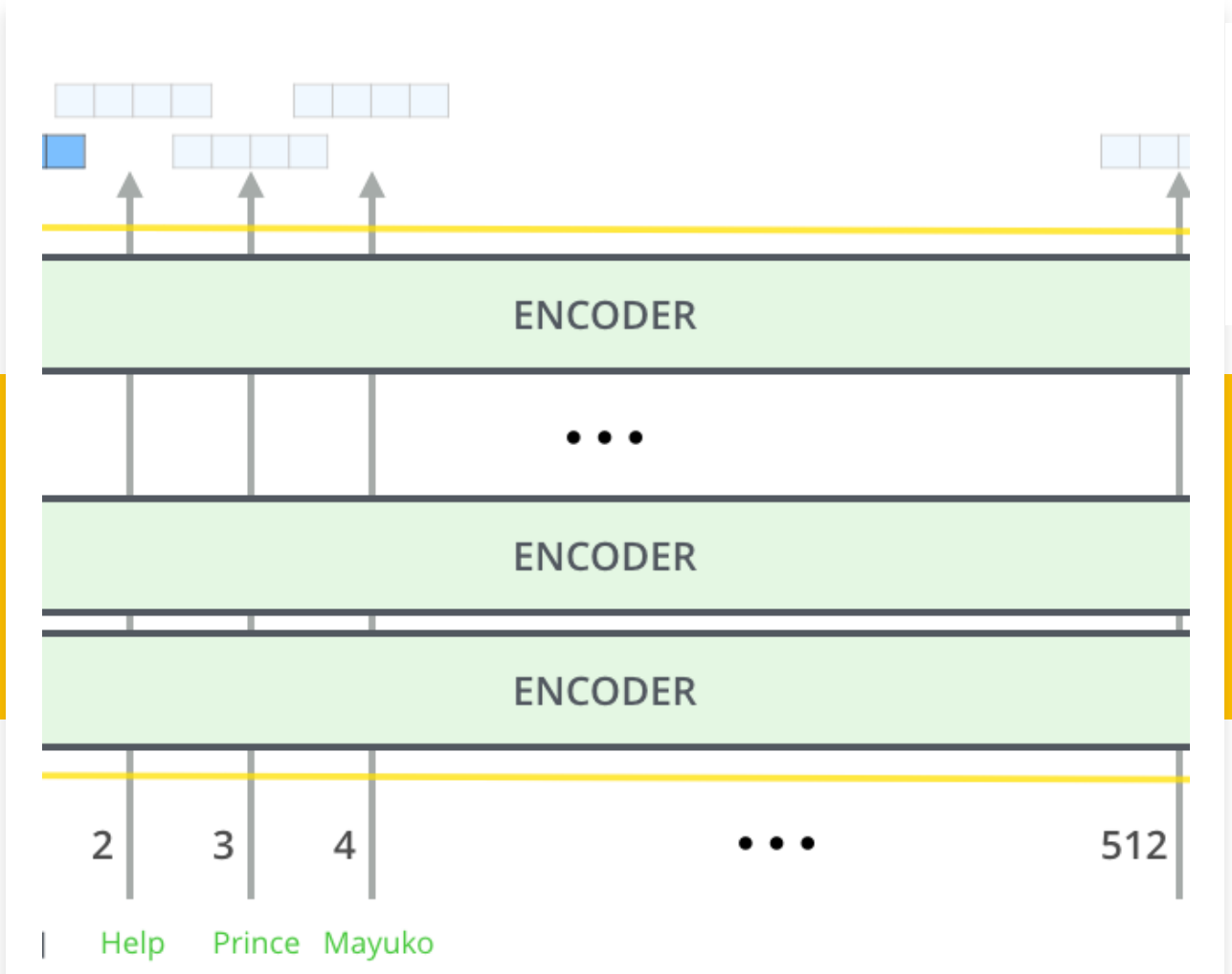


Based: Dr. Yangqiu Song's slides

Roadmap : f() on natural language

- Before Deep NLP (Pre 2012)
  - (BOW / LSI / Topic LDA)
- Word2Vec (2013-2016)
  - (GloVe/ FastText)
- Recurrent NN (2014-2016)
  - LSTM
  - Seq2Seq
- Attention / Self-Attention (2014 – now)
  - Attention
  - Transformer (self-attention, attention only)
  - BERT / XLNet/ GPT-2 / T5 …

BERT: **B**idirectional **Encoder** **R**epresentations from **Transformers**.





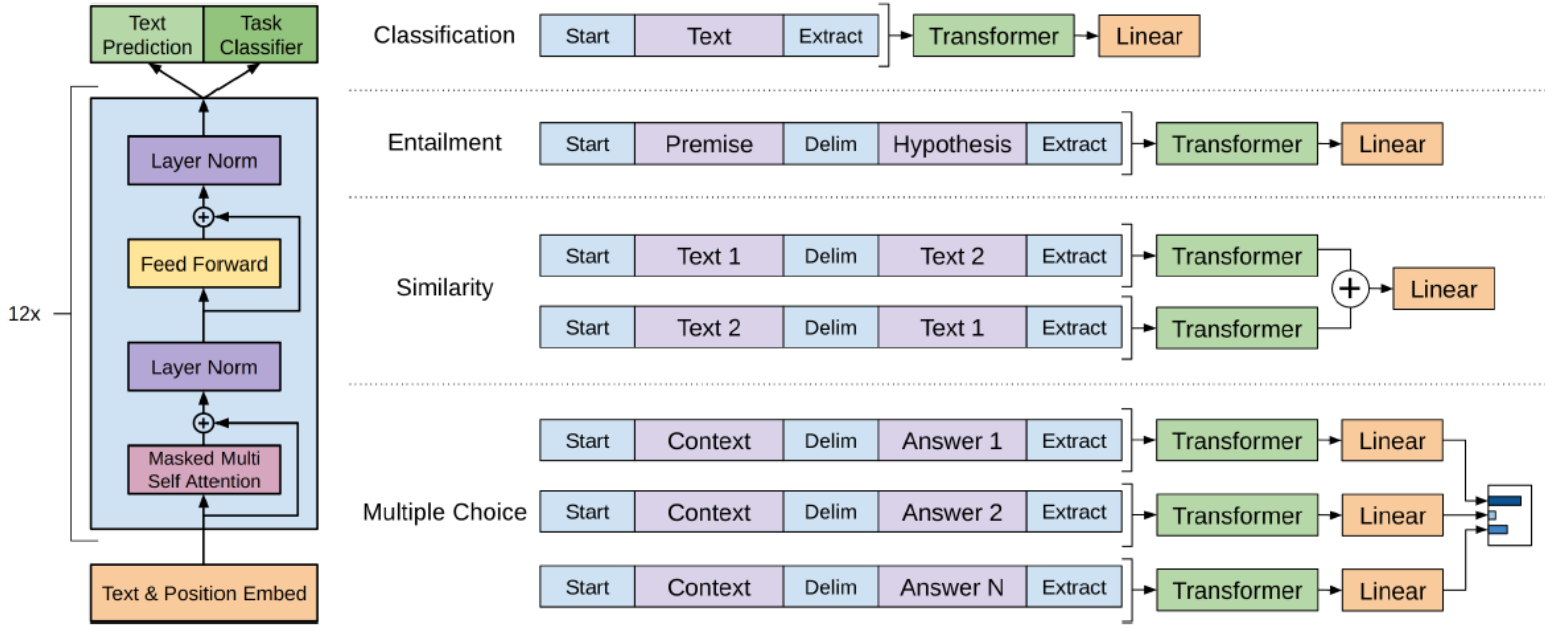
BERT's architecture is just a transformer's encoder stack.

Pre-trained transformer encoder for sentence embedding

Based: Dr. Yangqiu Song's slides

GPT: Generative pre-trained Transformer

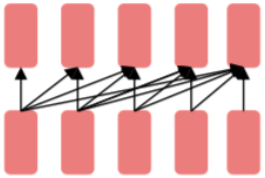
GPT1 (Radford et al. 2018) - Improving Language Understanding by Generative Pre-Training



- Pre-training: Maximize data likelihood as a product of conditional probabilities, trained on Books Corpus
- Predict each token based on the k tokens (the “context”) that came before

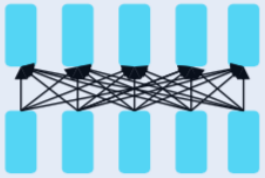
GPT 's architecture is just a transformer’s decoder stack.

Three types of transformer architectures



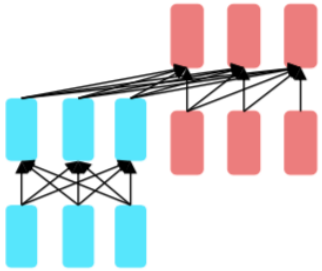
Decoders

- Nice to generate from; can't condition on future words
- **Examples:** GPT-2, GPT-3, LaMDA



Encoders

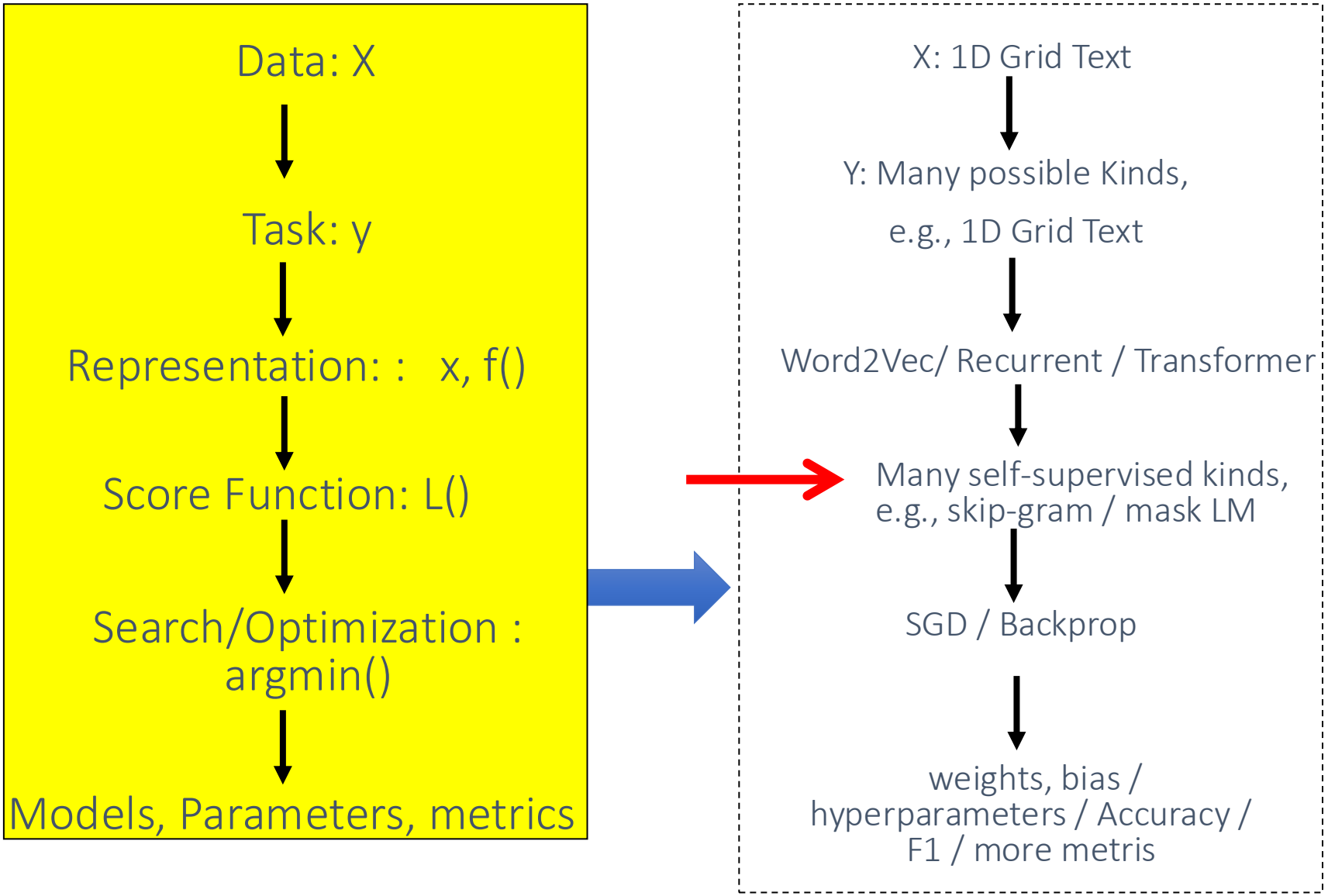
- Gets bidirectional context – can condition on future!
- Wait, how do we pretrain them?
- **Examples:** BERT and its many variants, e.g. RoBERTa

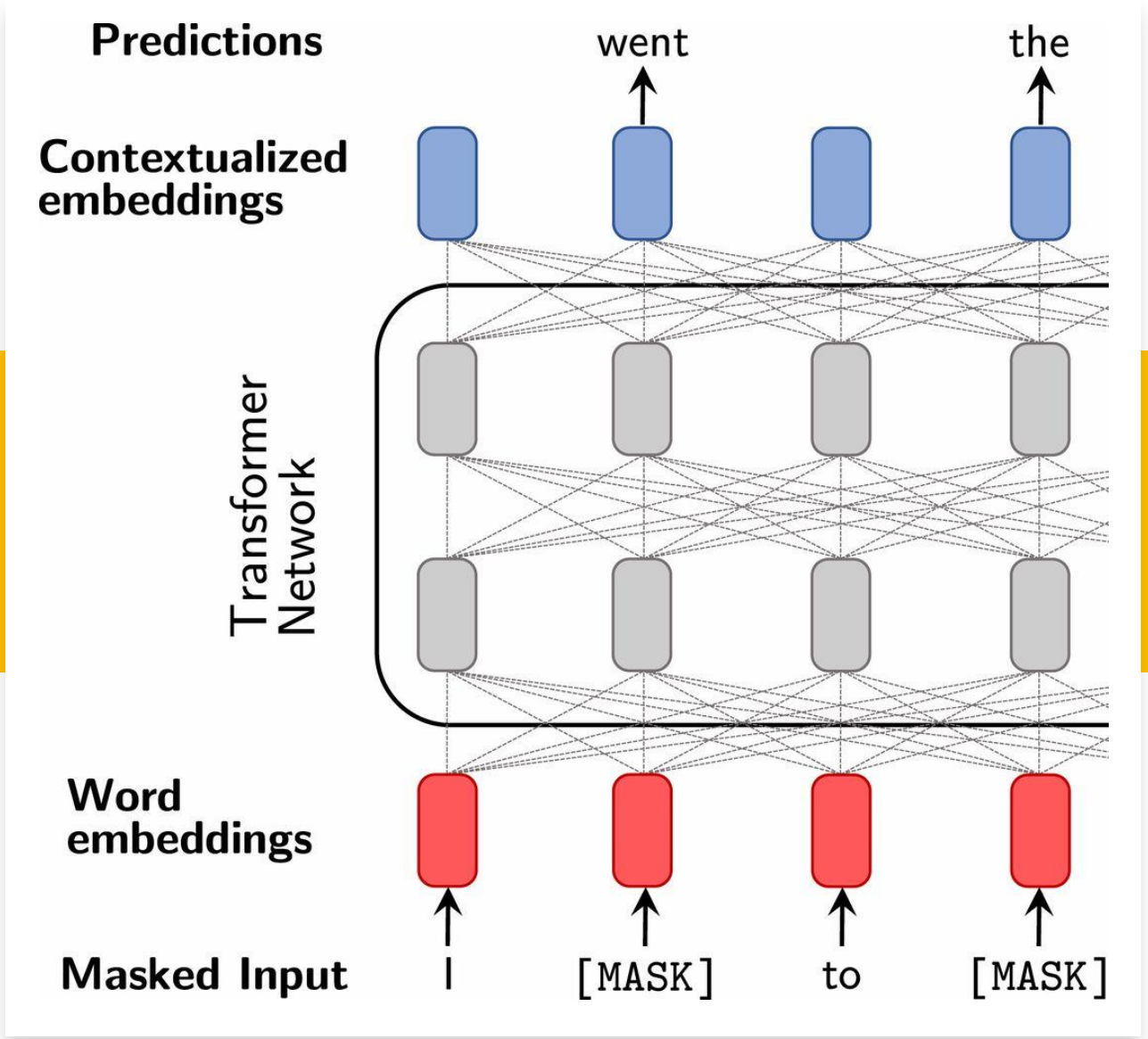


Encoder-Decoders

- Good parts of decoders and encoders?
- What's the best way to pretrain them?
- **Examples:** T5, Meena

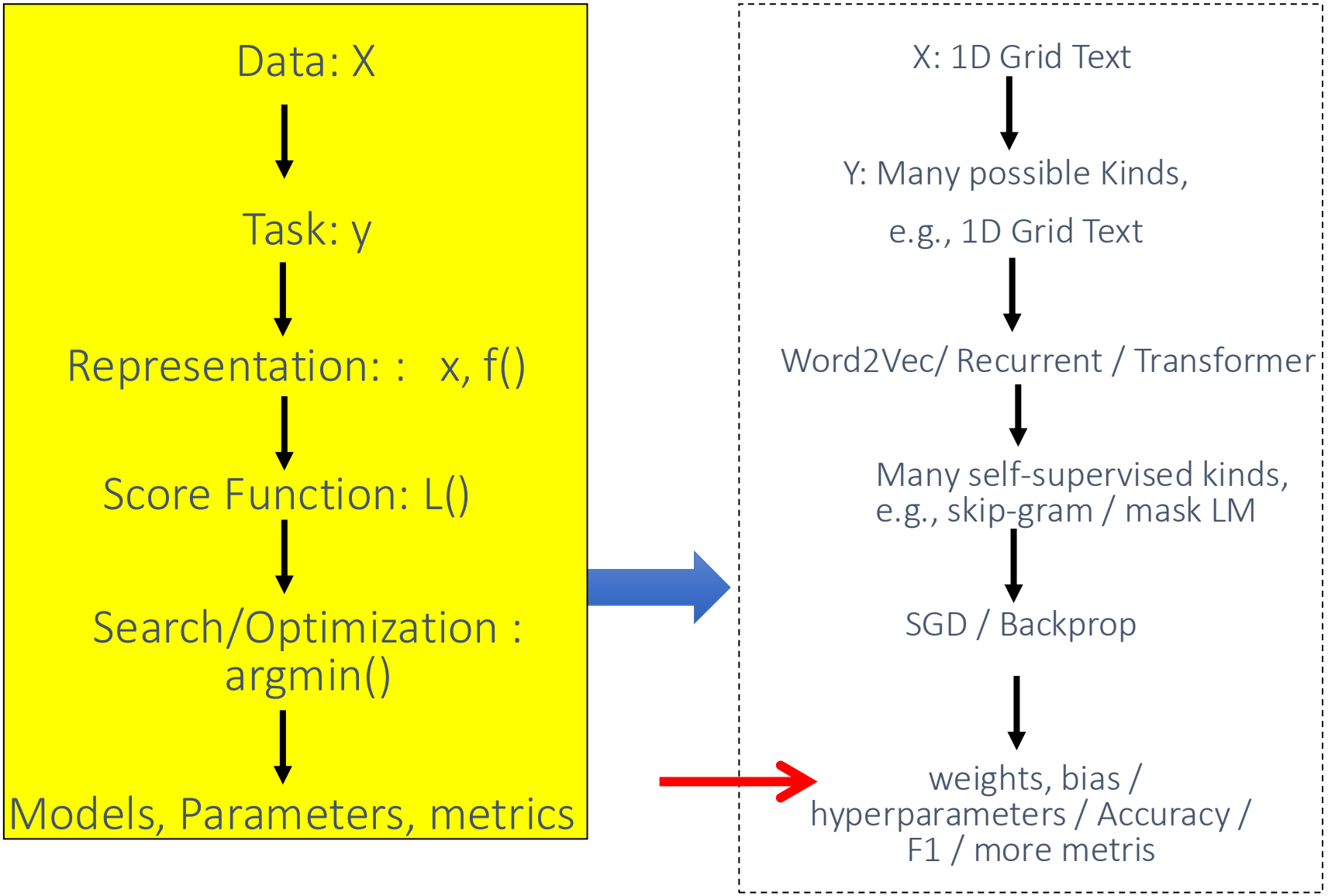
Today: Neural Network Models on 1D Grid / Language Data



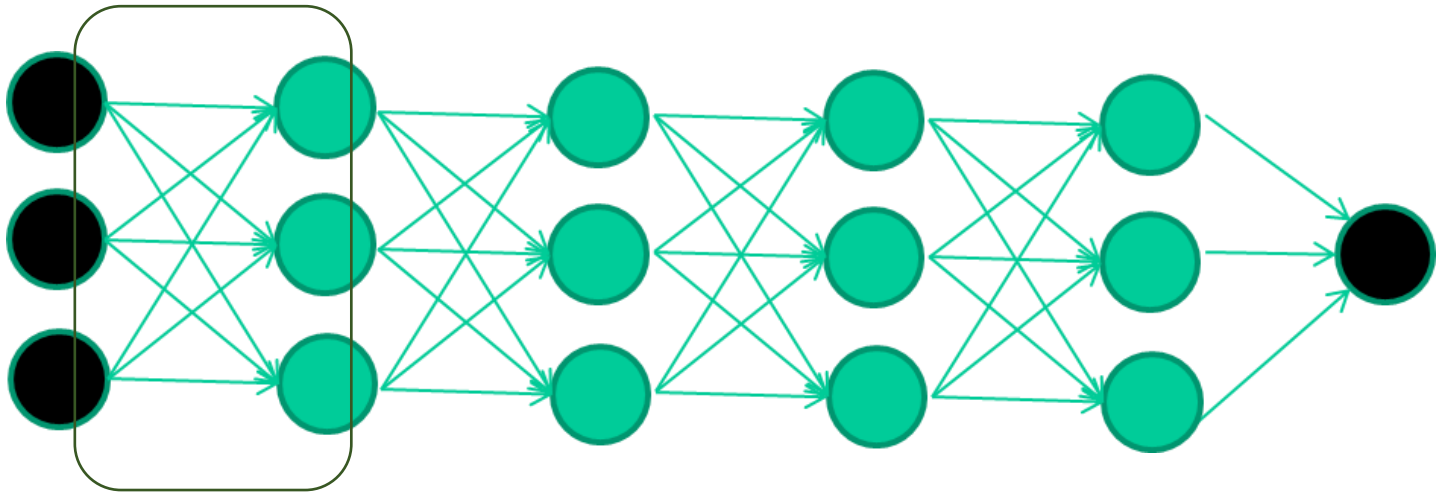


BERT is trained just like a skip-gram model.

Today: Neural Network Models on 1D Grid / Language Data



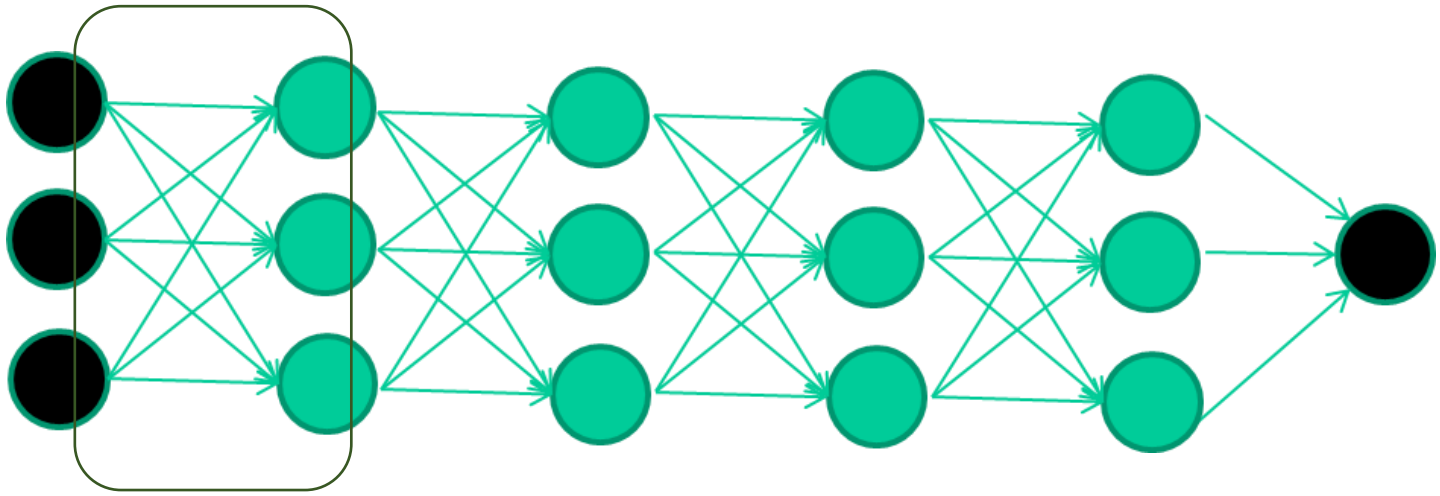
The new way to train multi-layer DNNs…



Train **this** layer first

https://www.macs.hw.ac.uk/~dwcorne/Teaching/introdl.ppt

The new way to train multi-layer DNNs…



Each layer can be trained well first with some types of self-supervised loss (e.g. reconstruction loss)

Basically, it is forced to learn good features that describe what comes from the previous layer

https://www.macs.hw.ac.uk/~dwcorne/Teaching/introdl.ppt

Unsupervised Pre-training vs. SFT

1. Unsupervised Pre-training
   - Use **unlabeled** data
   - Work bottom-up
     - Train hidden layer 1. Then fix its parameters.
     - Train hidden layer 2. Then fix its parameters.
     - …
     - Train hidden layer n. Then fix its parameters.
2. Supervised Fine-tuning
   - Use **labeled** data to train following “Idea #1”
   - Refine the features by back-propagation so that they become tuned to the end-task

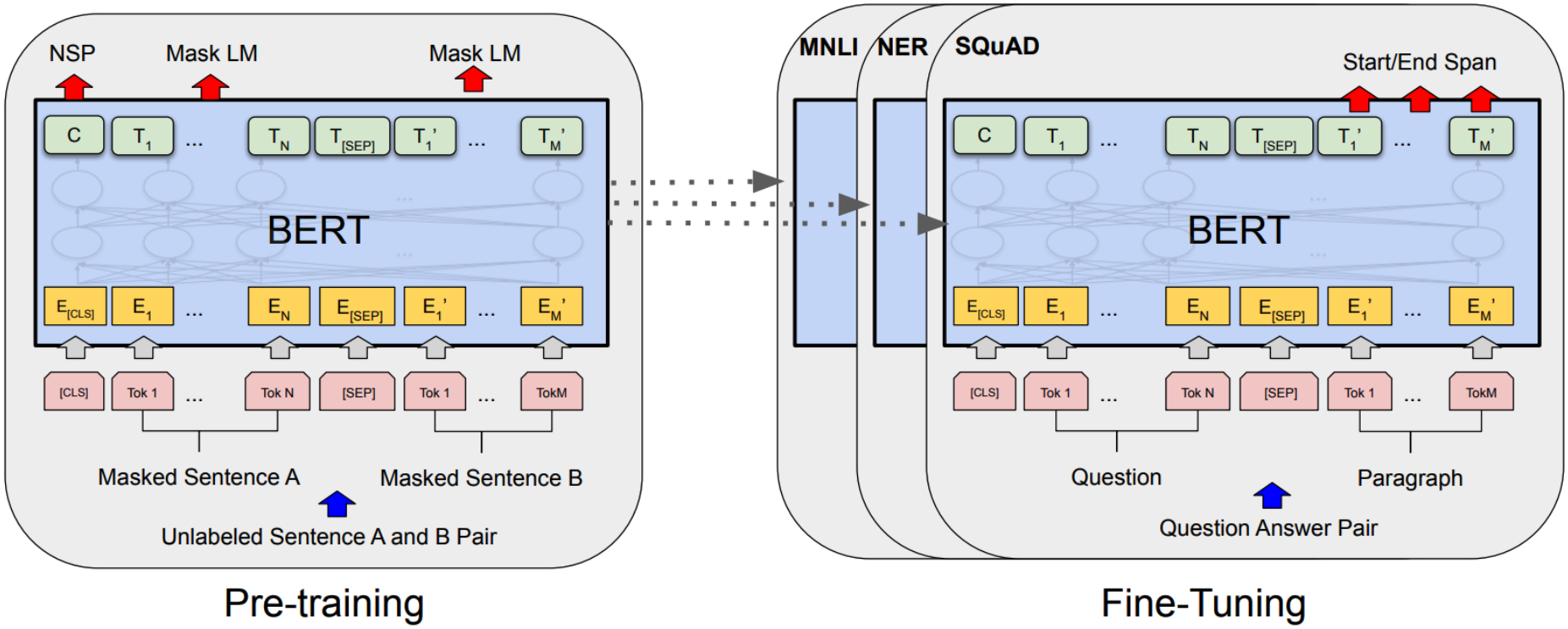
Pre-Training of DNN layer-layer Benefits:

- Layer-wise pre-training can help with:
  - Learn features at different levels of abstraction
  - DNN Model Training can be tricky due to Nonconvexity
  - DNN Model Training can be tricky due to Vanishing gradients

BERT ➔ Denoising Auto Encoder

- [MASK]: a unique token introduced in the training process to mask some tokens
- Predict masked tokens based on their context information,
- Pre-train and fine-tune

- Intuition: representation should be robust to the introduction of noise
  - Masked Language Model (MLM)



google-research

ALBERT: A lite BERT (2019, Lan et al.)

- proposes Sentence Order Prediction (SOP) task to replace Next Sentence Prediction (NSP)
- in NSP, the negative next sentence is sampled from other passages that may have different topics with the current one, turning the NSP into a far easier topic model problem.
- in SOP, two sentences that exchange their position are regarded as a negative sample, making the model concentrate on the coherence of the semantic meaning.

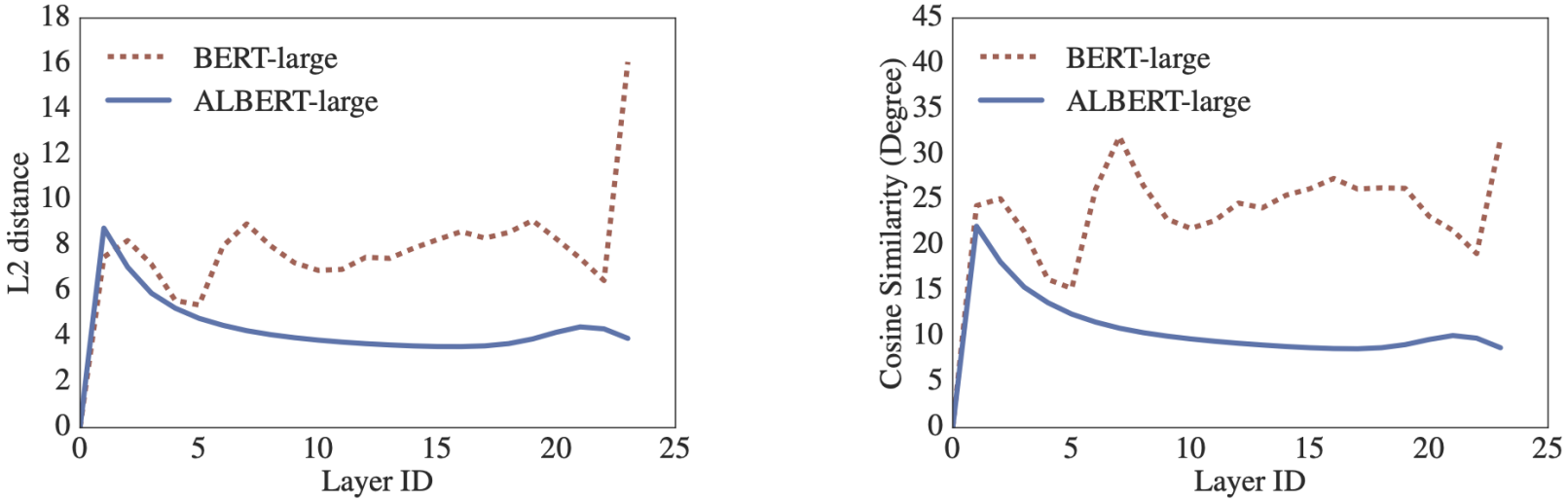
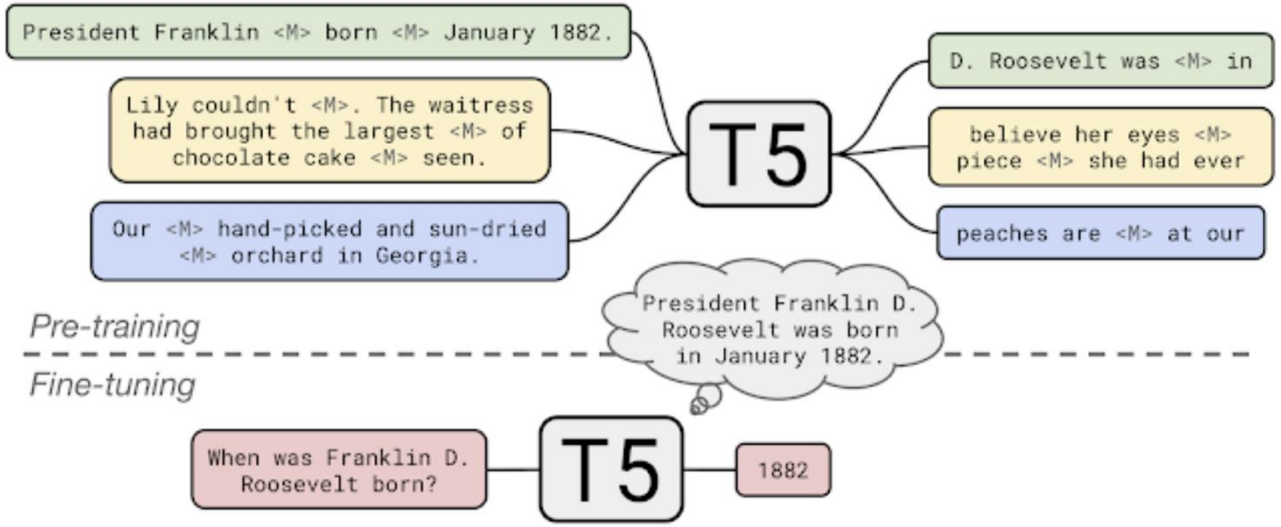


Figure 1: The L2 distances and cosine similarity (in terms of degree) of the input and output embedding of each layer for BERT-large and ALBERT-large.

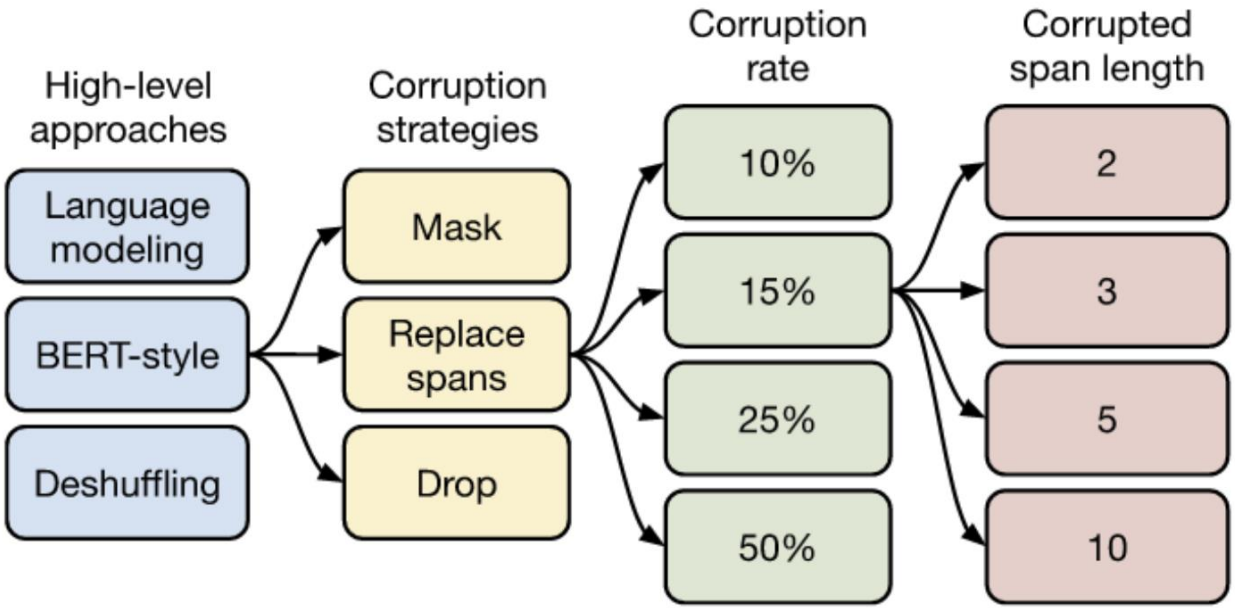
Based: Dr. Yangqiu Song's slides

google-research

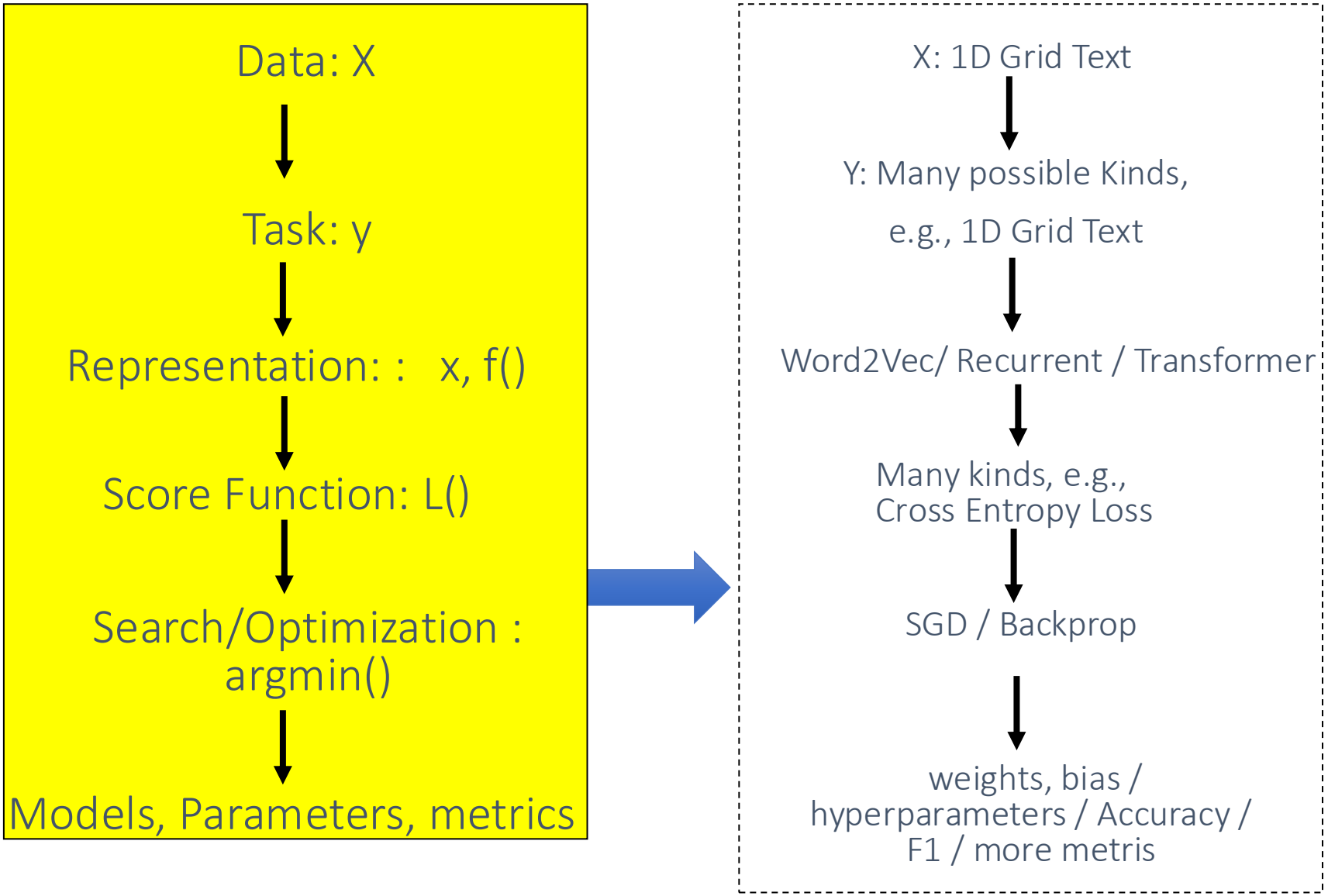
T5: Even more noise



During pre-training, T5 learns to fill in dropped-out spans of text (denoted by <M>) from documents in C4. To apply T5 to closed-book question answer, we fine-tuned it to answer questions without inputting any additional information or context. This forces T5 to answer questions based on "knowledge" that it internalized during pre-training.



Today Recap: Neural Network Models on 1D Grid / Language Data



Recap : f() on natural language inputs

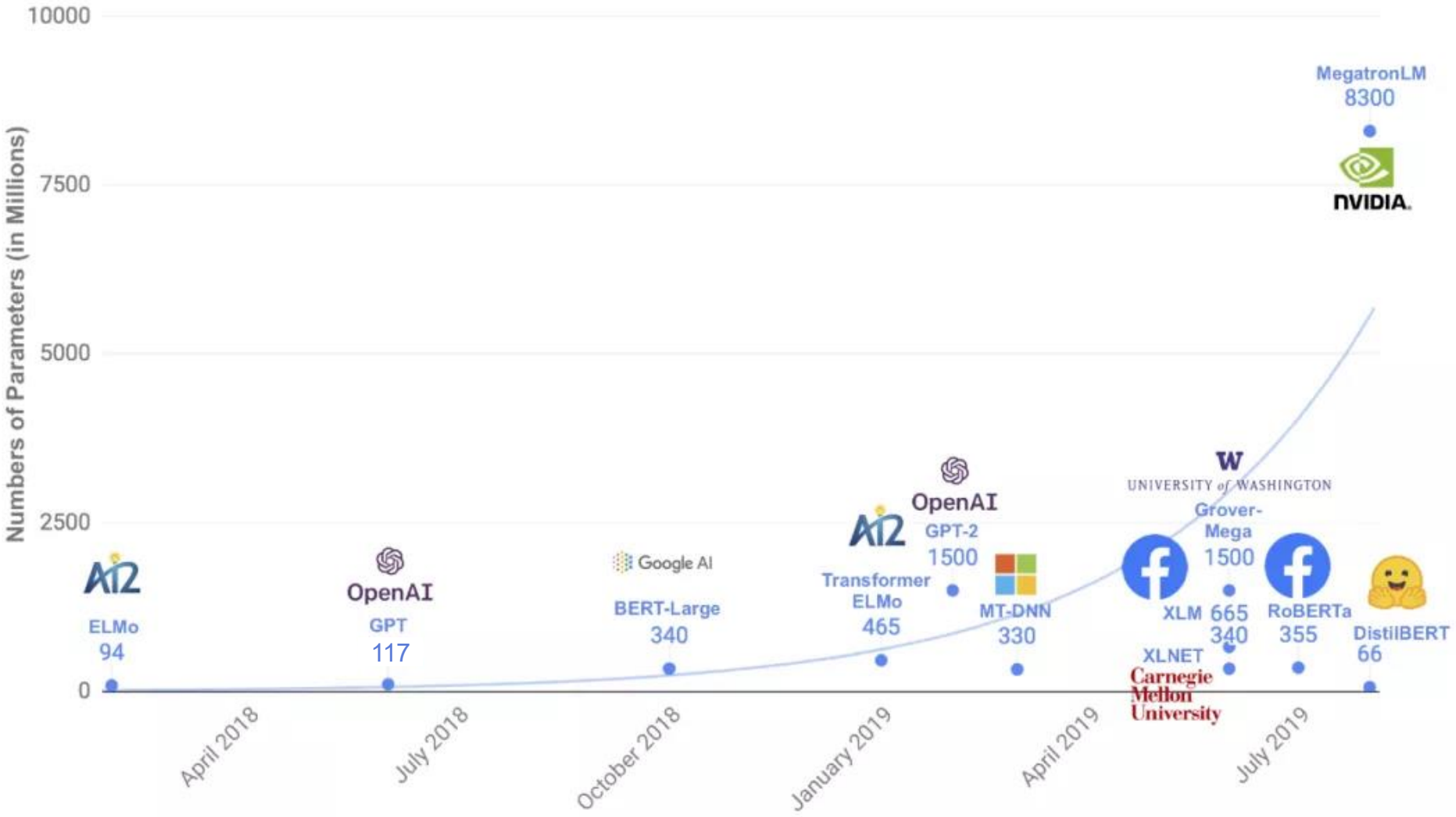
- Before Deep NLP (Pre 2012)
  - (BOW / LSI / Topic LDA)
- Word2Vec (2013-2016)
  - (GloVe/ FastText)
- Recurrent NN (2014-2016)
  - LSTM
  - Seq2Seq
- Attention / Self-Attention (2014 – now)
  - Attention
  - Transformer (self-attention, attention only)
  - BERT / XLNet/ GPT-2 / T5 …

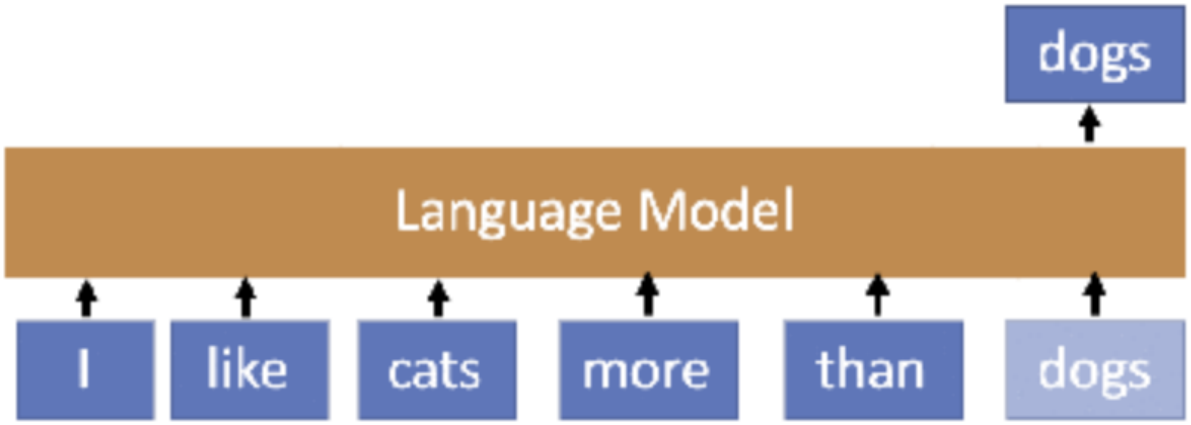
Takeaways and what's next

- Seq2seq: an encoder reads, a decoder writes; one fixed vector is the bottleneck
- Attention: $c_t = \Sigma_i \alpha_{i,t} h_i$ gives a new context at each output step
- Self-attention: a sequence attends to itself, so no recurrence is needed
- Transformer: BERT keeps the encoders, GPT keeps the decoders
- Pretraining: the text supplies its own labels, then we fine-tune

References

- Dr. Yann Lecun's deep learning tutorials
- Dr. Li Deng's ICML 2014 Deep Learning Tutorial
- Dr. Kai Yu's deep learning tutorial
- Dr. Rob Fergus' deep learning tutorial
- Prof. Nando de Freitas' slides
- Olivier Grisel's talk at Paris Data Geeks / Open World Forum
- Hastie, Trevor, et al. *The elements of statistical learning*. Vol. 2. No. 1. New York: Springer, 2009.
- Dr. Hung-yi Lee's CNN slides
- NIPS 2017 DL Trend Tutorial


10000
7500
5000
2500
0
Numbers of Parameters (in Millions)
April 2018
July 2018
October 2018
January 2019
April 2019
July 2019
ELMo
94
OpenAI
GPT
117
Google AI
BERT-Large
340
Transformer
ELMo
465
OpenAI
GPT-2
1500
MT-DNN
330
XLM
665
XLNET
340
Carnegie
Mellon
University
UNIVERSITY of WASHINGTON
Grover-
Mega
1500
RoBERTa
355
DistilBERT
66
MegatronLM
8300
NVIDIA.



The prediction scheme for a traditional language model. Shaded words are provided as input to the model while unshaded words are masked out.

Autoregressive Models

$$P(x;\theta) = \prod_{n=1}^{N} P(x_n | x_{<n};\theta)$$

- Each factor can be parametrized by θ, which can be shared.
- The variables can be arbitrarily ordered and grouped, as long as the ordering and grouping is consistent.

Adapt from From NIPS 2017 DL Trend Tutorial