PAT 498/598 (Fall 2024)

Special Topics:
Generative AI for Music and Audio Creation

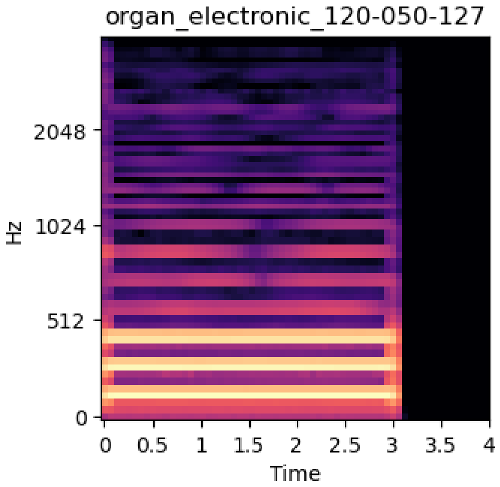
Lecture 8: RNNs, LSTMs & Transformers

Instructor: Hao-Wen Dong

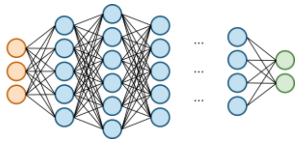
SCHOOL OF MUSIC, THEATRE & DANCE
PERFORMING ARTS TECHNOLOGY
UNIVERSITY OF MICHIGAN

Assignment 2: Musical Note Classification using CNNs

- Train a CNN that can classify audio files into their **instrument families**
  - **Input**: 64x64 mel spectrogram
  - **Output**: 11 instrument classes
  - Using the **NSynth** dataset (Engel et al., 2017)



→



→

Bass
Brass
Flute
Guitar
Keyboard
Mallet
Organ
Reed
String
Synth Lead
Vocal

Jesse Engel, Cinjon Resnick, Adam Roberts, Sander Dieleman, Douglas Eck, Karen Simonyan, and Mohammad Norouzi, "Neural Audio Synthesis of Musical Notes with WaveNet Autoencoders," *ICML*, 2017.

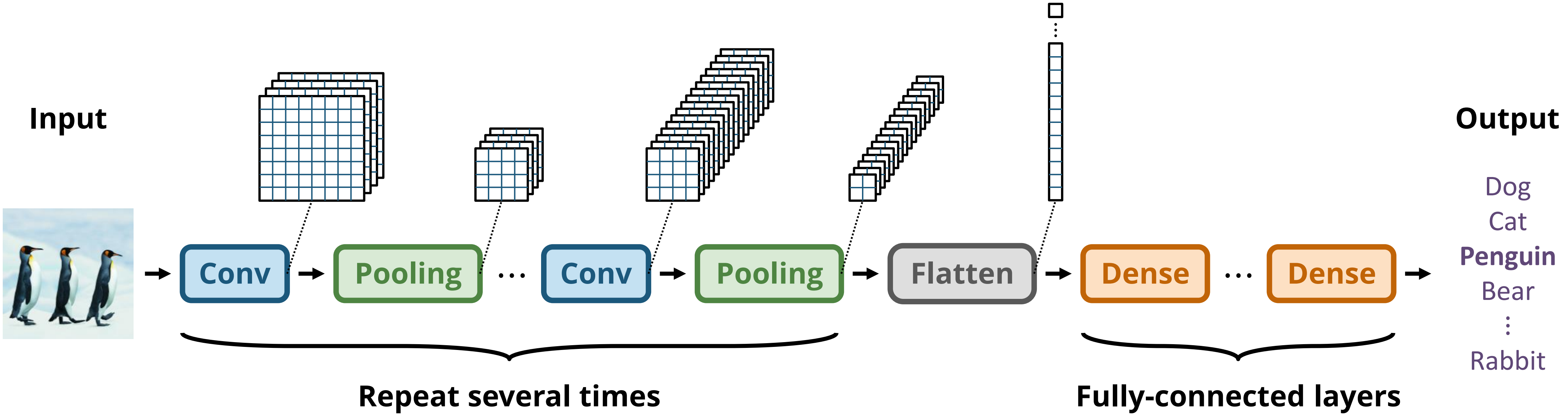
Assignment 2: Musical Note Classification using CNNs

- Instructions will be released on Gradescope
- Due at **11:59pm ET** on **October 7**
- Late submissions: **3 point deducted per day**

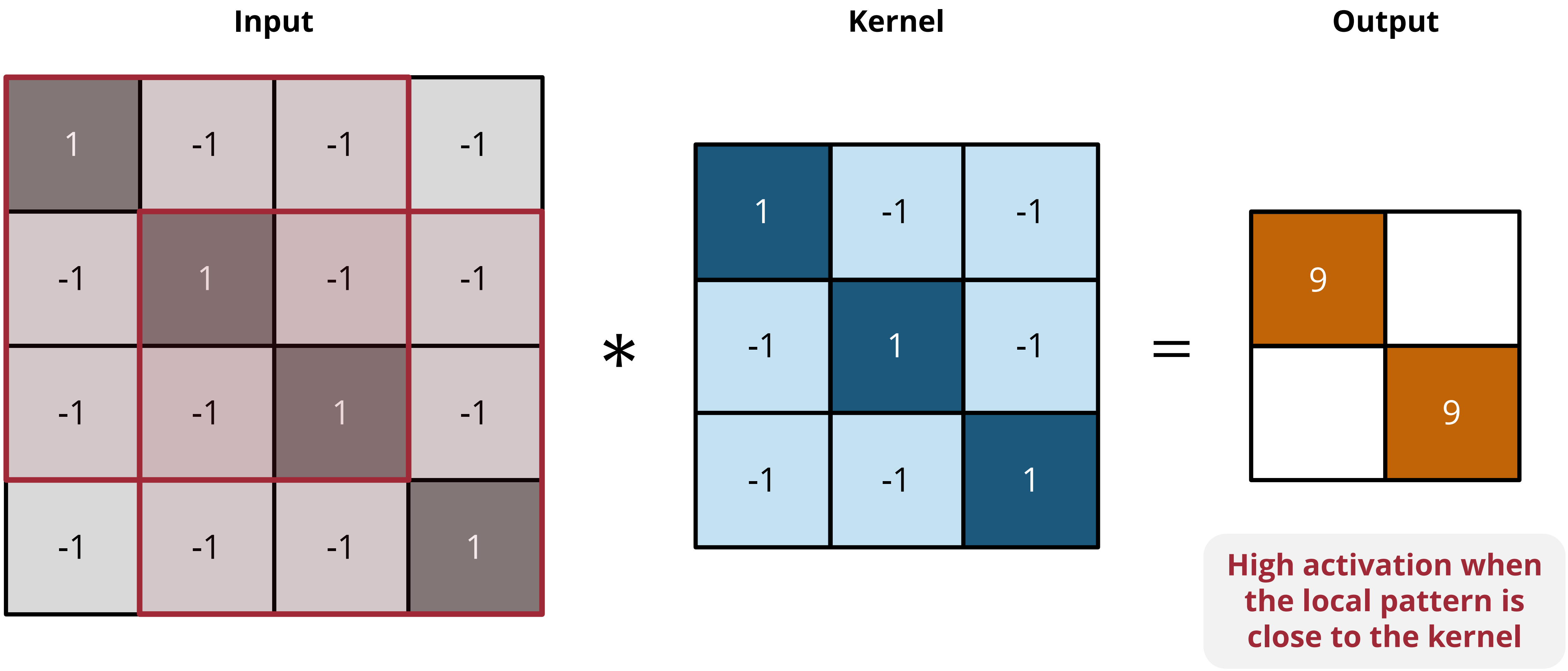
Great Lakes

- **Great Lakes** is a high-performance computing cluster at U-M
- You will be provided **3000 CPU hours** (**~400 GPU hours**)
- Before you access Great Lakes, you'll need to first **create an HPC login**!
- **U-M VPN** is required to access the web portal off-campus

(Recap) Convolutional Neural Network (CNNs)



(Recap) 2D Convolution

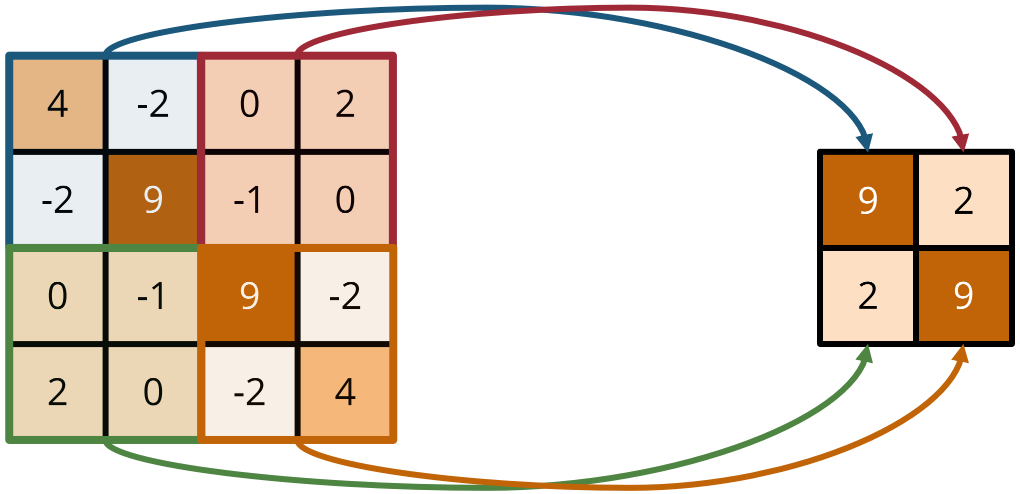


(Recap) 2D Convolution

$$
\underset{\textbf{Input}}{\begin{bmatrix} 1 & -1 & -1 & -1 \\ -1 & 1 & -1 & -1 \\ -1 & -1 & 1 & -1 \\ -1 & -1 & -1 & 1 \end{bmatrix}} * \underset{\textbf{Kernel}}{\begin{bmatrix} 1 & -1 & -1 \\ -1 & 1 & -1 \\ -1 & -1 & 1 \end{bmatrix}} = \underset{\textbf{Output}}{\begin{bmatrix} \ & -1 \\ -1 & \ \end{bmatrix}}
$$

Low activation when the local pattern differs from the kernel

(Recap) Max Pooling Layer



Downsample and keep the strongest activation in each block

(Recap) Learned CNN Kernels in a Trained AlexNet

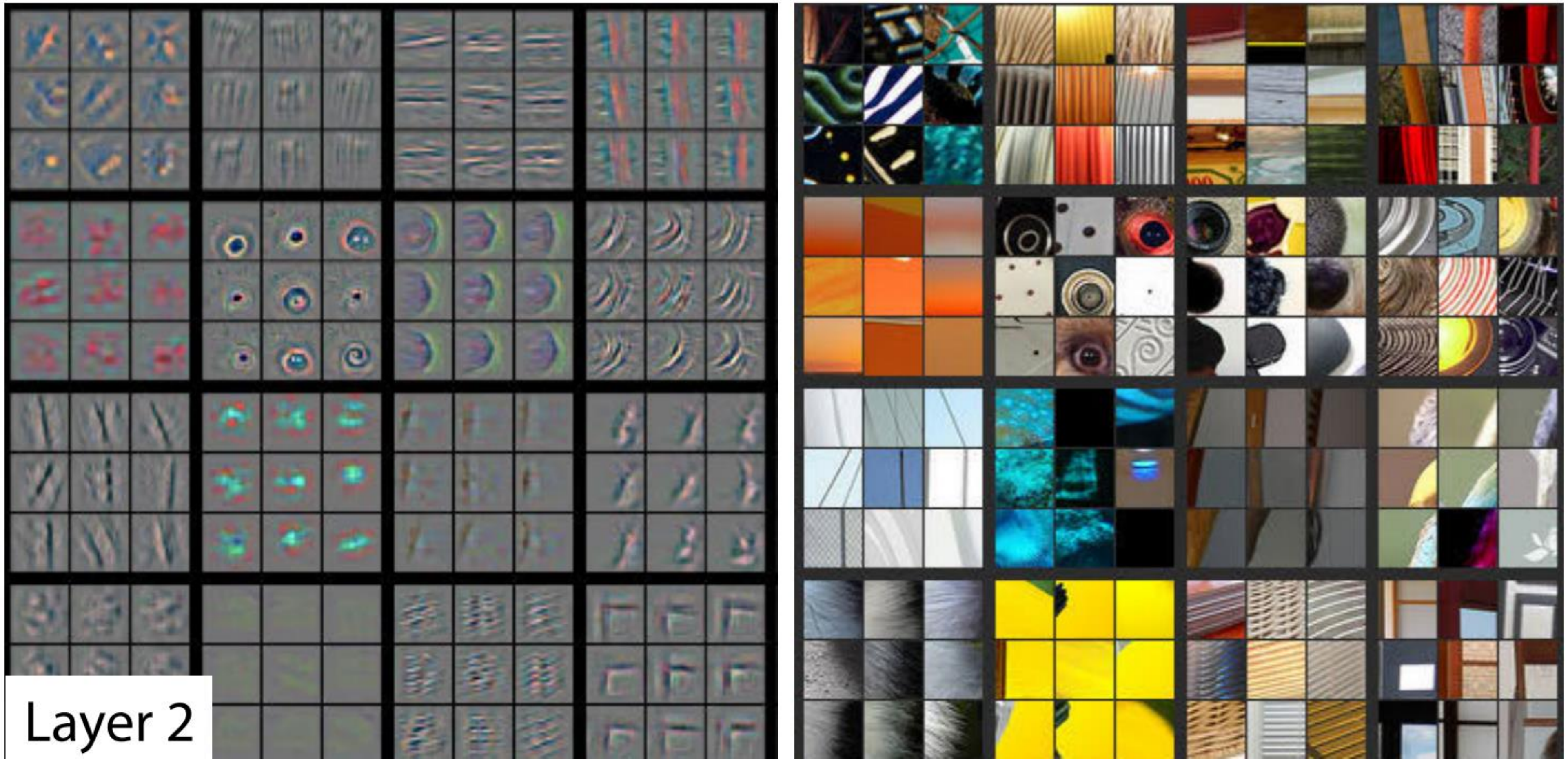
Layer 1

Learned CNN kernels

Top activations

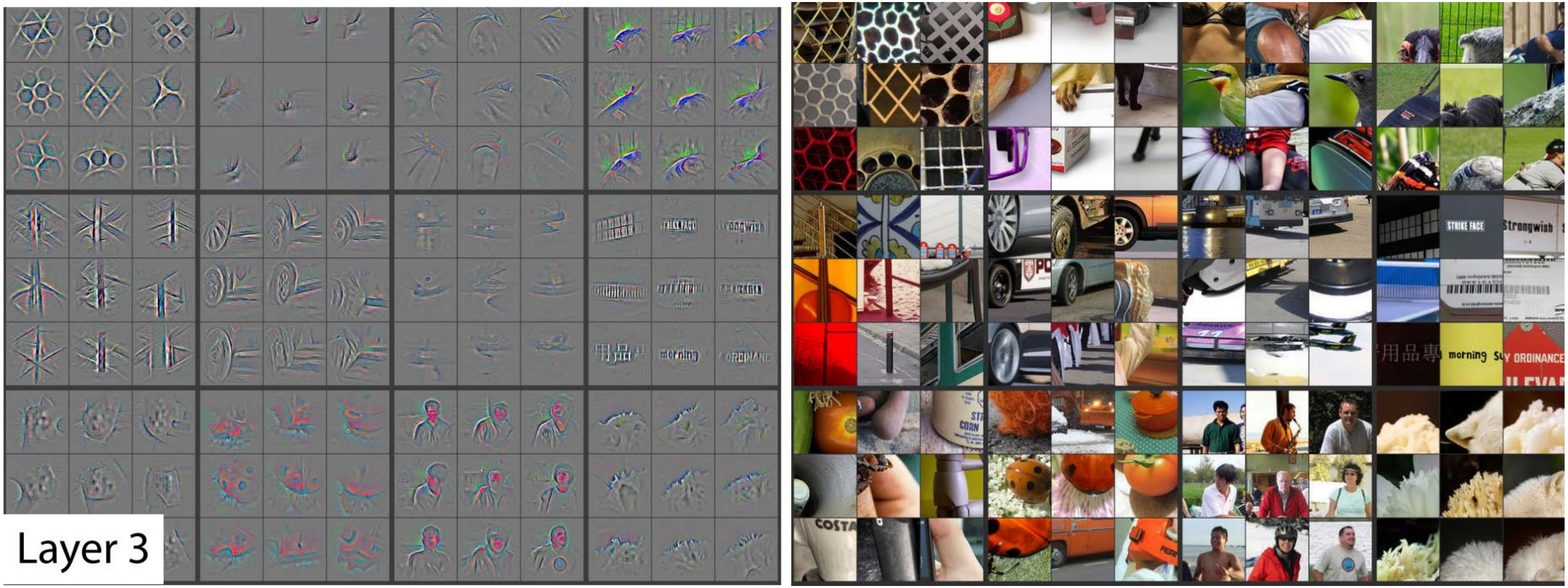
Matthew D. Zeiler and Rob Fergus, "Visualizing and Understanding Convolutional Networks," *ECCV*, 2014.

(Recap) Learned CNN Kernels in a Trained AlexNet



Matthew D. Zeiler and Rob Fergus, "Visualizing and Understanding Convolutional Networks," *ECCV*, 2014.

(Recap) Learned CNN Kernels in a Trained AlexNet

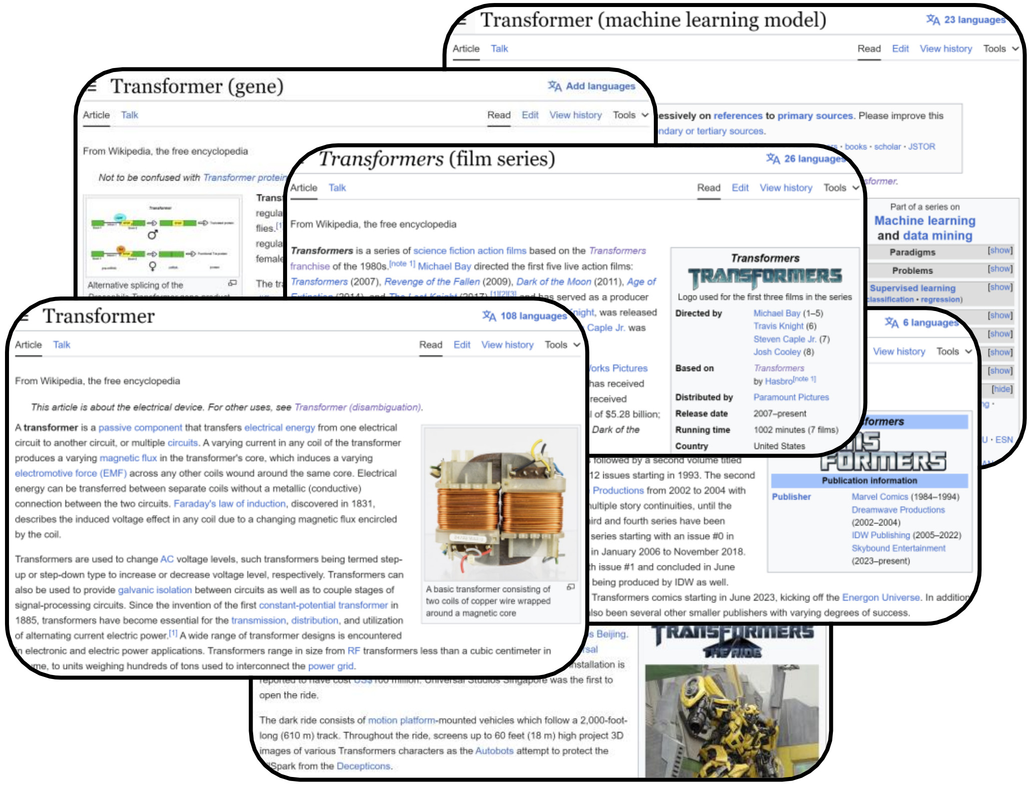


Matthew D. Zeiler and Rob Fergus, "Visualizing and Understanding Convolutional Networks," *ECCV*, 2014.

Language Models

Language Models

- Predicting the next word **given the past sequence of words**



A transformer is a ________

electrical device

fiction character

deep learning model

family of genes

type of food

musical instrument

Language Models (Mathematically)

- A class of machine learning models that learn the next word probability

$$P(\, x_i \mid \underbrace{x_1, x_2, \ldots, x_{i-1}}\,)$$

Next word Previous words

$P($ electrical | A transformer is a $)$ ↑

$P($ character | A transformer is a $)$ ↑

$P($ gene | A transformer is a $)$ ↑

$P($ model | A transformer is a $)$ ↑

$P($ food | A transformer is a $)$ ↓

$P($ musical | A transformer is a $)$ ↓

Language Models – Generation

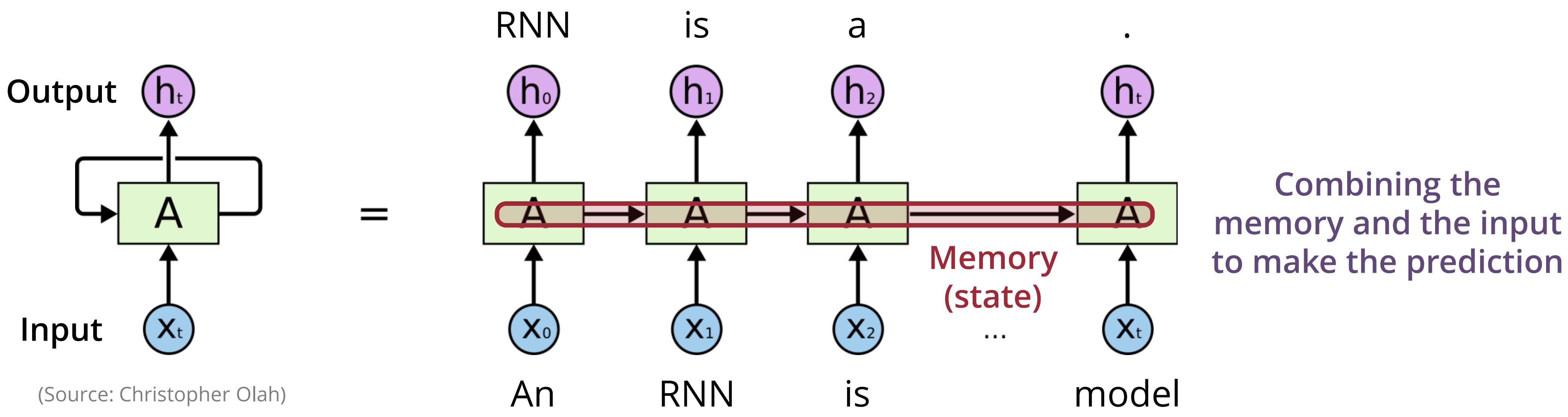
- How do we generate a new sentence using a trained language model?

A transformer is a	→	Model	→	deep
A transformer is a deep	→	Model	→	learning
A transformer is a deep learning	→	Model	→	model
A transformer is a deep learning model	→	Model	→	introduced
A transformer is a deep learning model introduced	→	Model	→	in
A transformer is a deep learning model introduced in	→	Model	→	2017

Recurrent Neural Networks (RNNs)

What is an RNN (Recurrent Neural Network)?

- A type of neural networks that have **loops**
- Widely used for **modeling sequences** (e.g., in natural language processing)

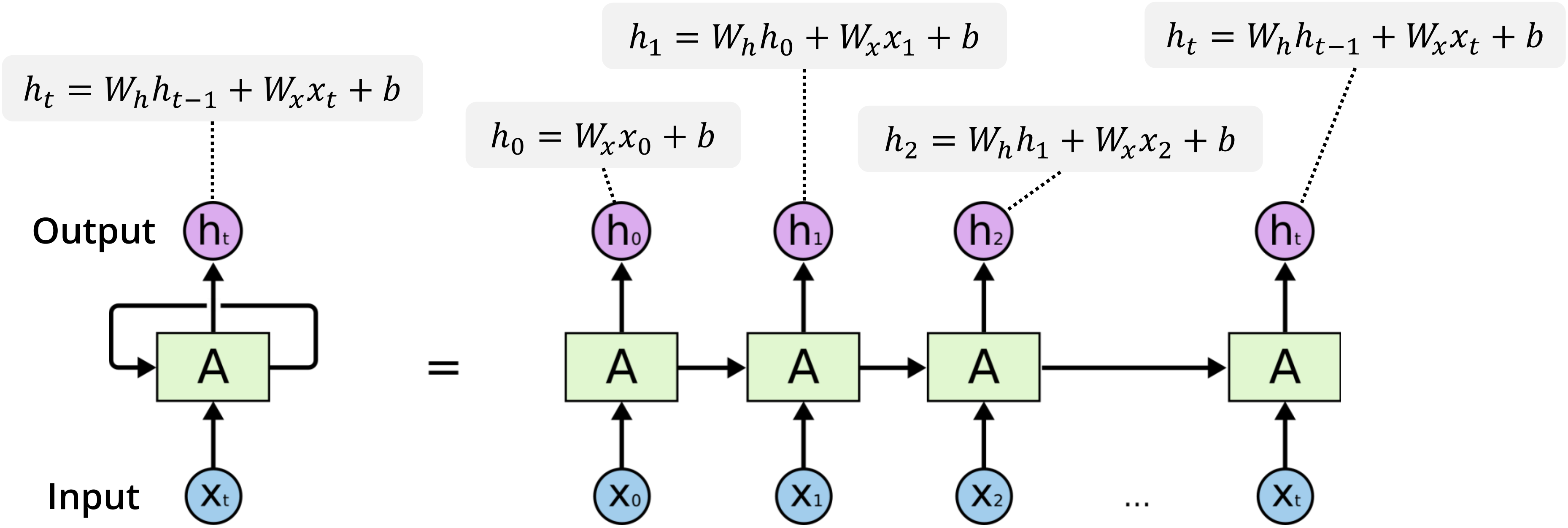


(Source: Christopher Olah)

colah.github.io/posts/2015-08-Understanding-LSTMs/

Vanilla RNNs

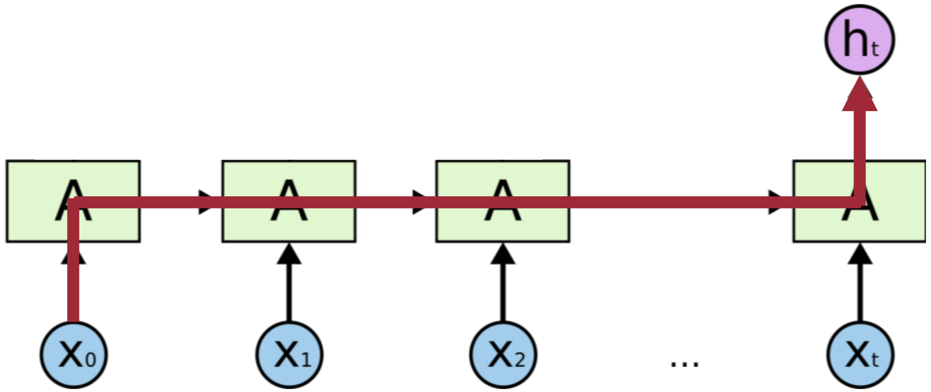
- The simplest form of RNNs
- LSTMs and GRUs are also RNNs



(Source: Christopher Olah)

Backpropagation Through Time

- An RNN is essentially a **very deep neural network**



$$h_t = W_h h_{t-1} + W_x x_t + b$$

$$h_t = W_h(W_h h_{t-2} + W_x x_{t-1} + b) + W_x x_t + +b$$

$$\vdots$$

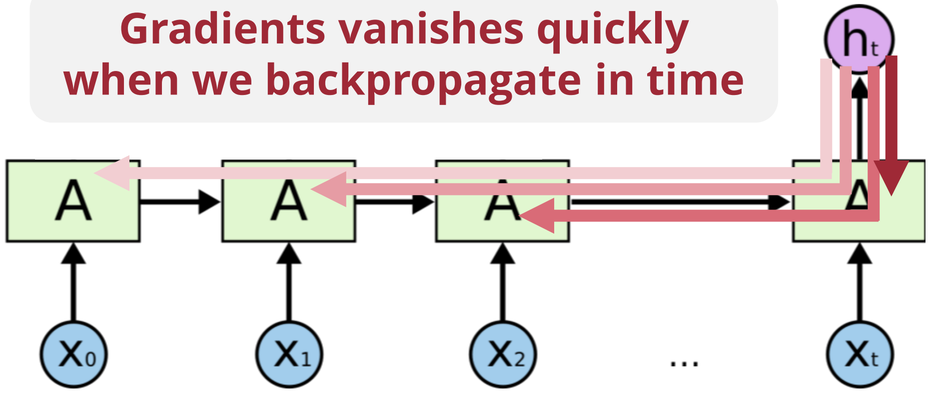
$$h_t = W_h(W_x x_{t-1} + W_h(\cdots \ W_h h_0 + W_x x_1 + b \ \cdots) + b) + W_x x_t + +b$$

colah.github.io/posts/2015-08-Understanding-LSTMs/

Vanishing Gradients

- An RNN is essentially a **very deep neural network**

Gradients vanishes quickly when we backpropagate in time



All the layers share the same weight matrix

Can still train the model without deeper gradients

Why bother?

$$h_t = W_h h_{t-1} + W_x x_t + b$$

$$h_t = W_h(W_h h_{t-2} + W_x x_{t-1} + b\,) + W_x x_t + +b$$

$$\vdots$$

$$h_t = W_h(W_x x_{t-1} + W_h(\,\cdots\; W_h h_0 + W_x x_1 + b\;\cdots\,) + b\,) + W_x x_t + +b$$

colah.github.io/posts/2015-08-Understanding-LSTMs/

Long Short-Term Memory (LSTMs)

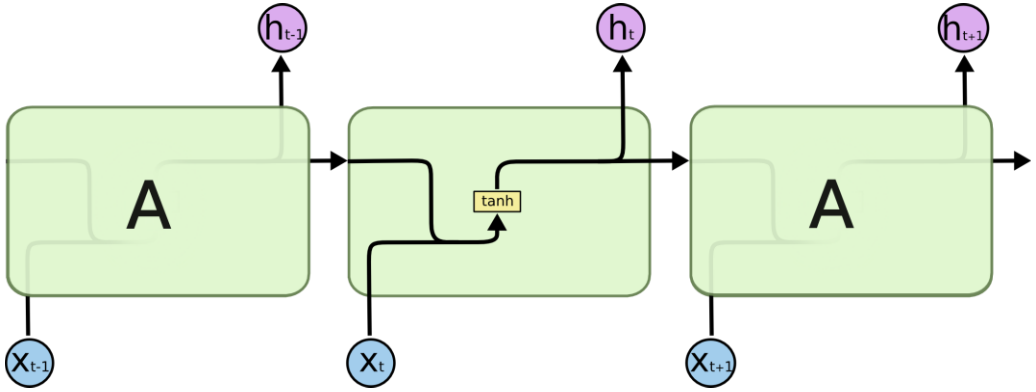
Vanilla RNNs vs LSTMs (Long Short-Term Memory)

Vanilla RNN

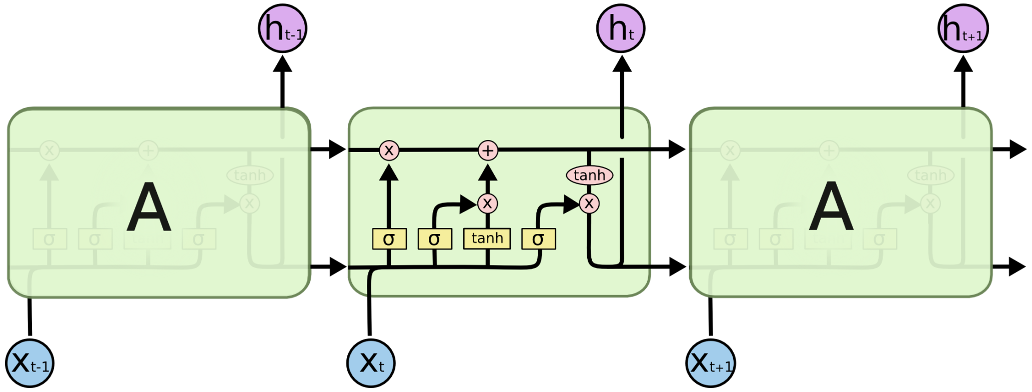
- Simplest form of RNNs
- Limited long-term memory

LSTM

- Improved memory module
- Better long-term memory

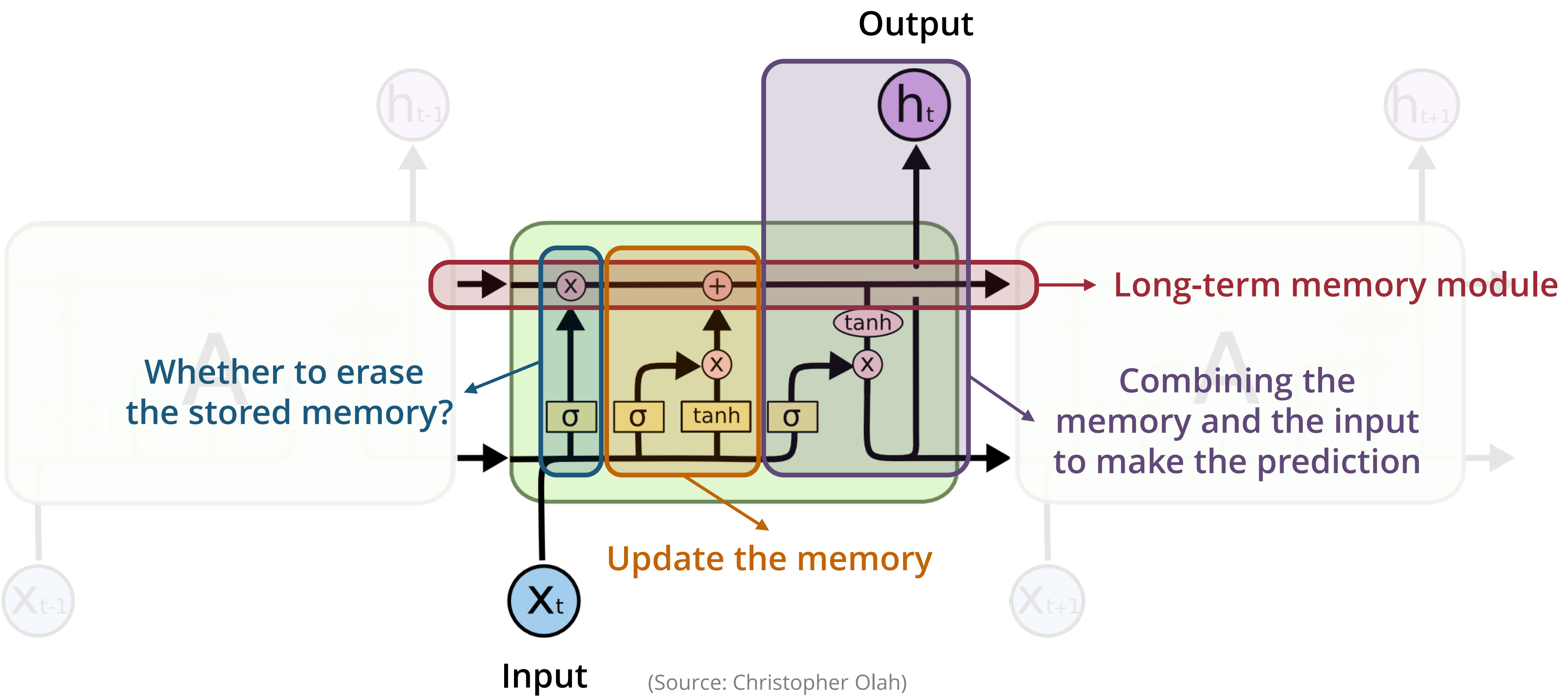


(Source: Christopher Olah)



(Source: Christopher Olah)

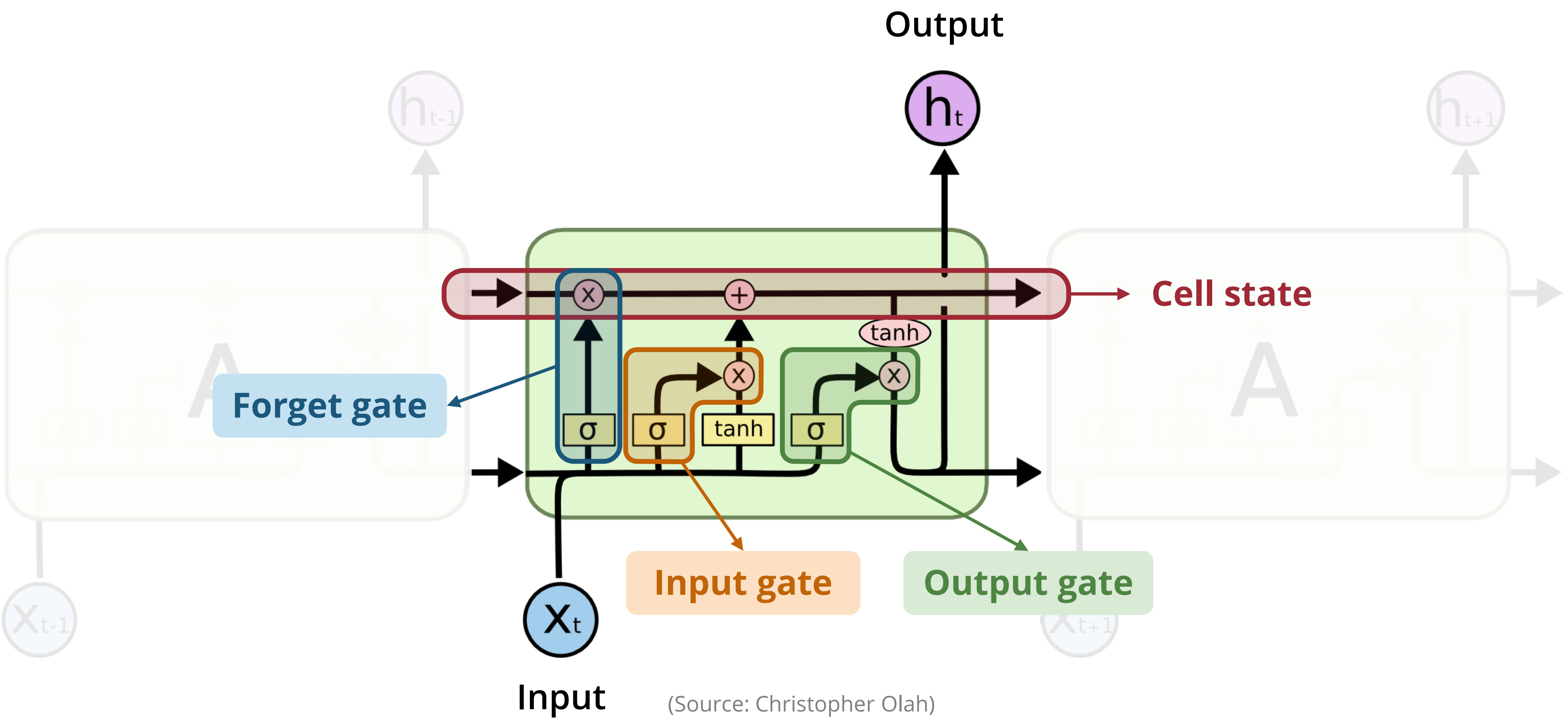
Demystifying LSTMs



(Source: Christopher Olah)

colah.github.io/posts/2015-08-Understanding-LSTMs/

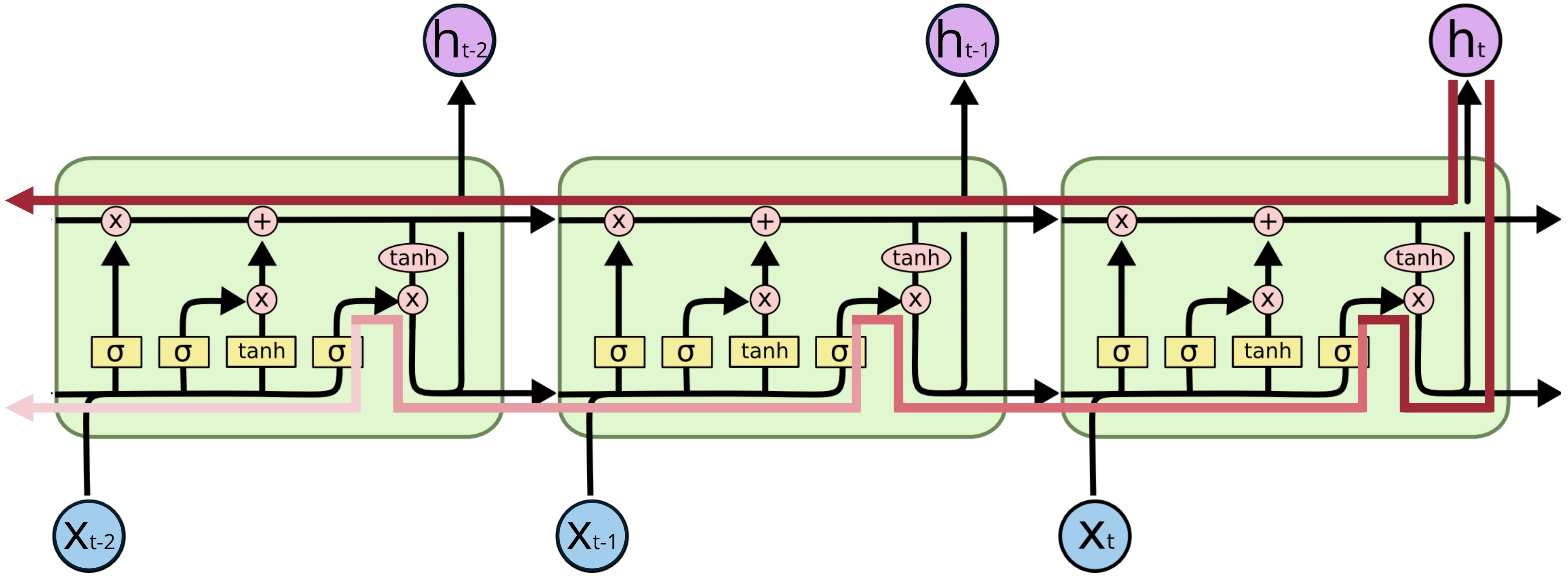
Demystifying LSTMs



(Source: Christopher Olah)

colah.github.io/posts/2015-08-Understanding-LSTMs/

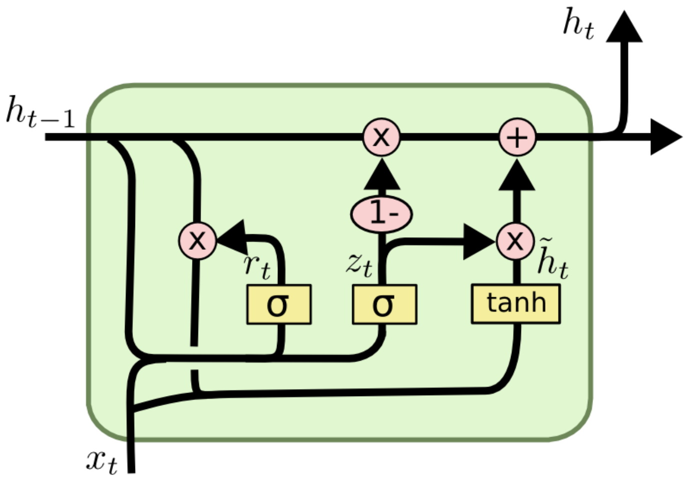
How can LSTMs Help Alleviate Vanishing Gradients?



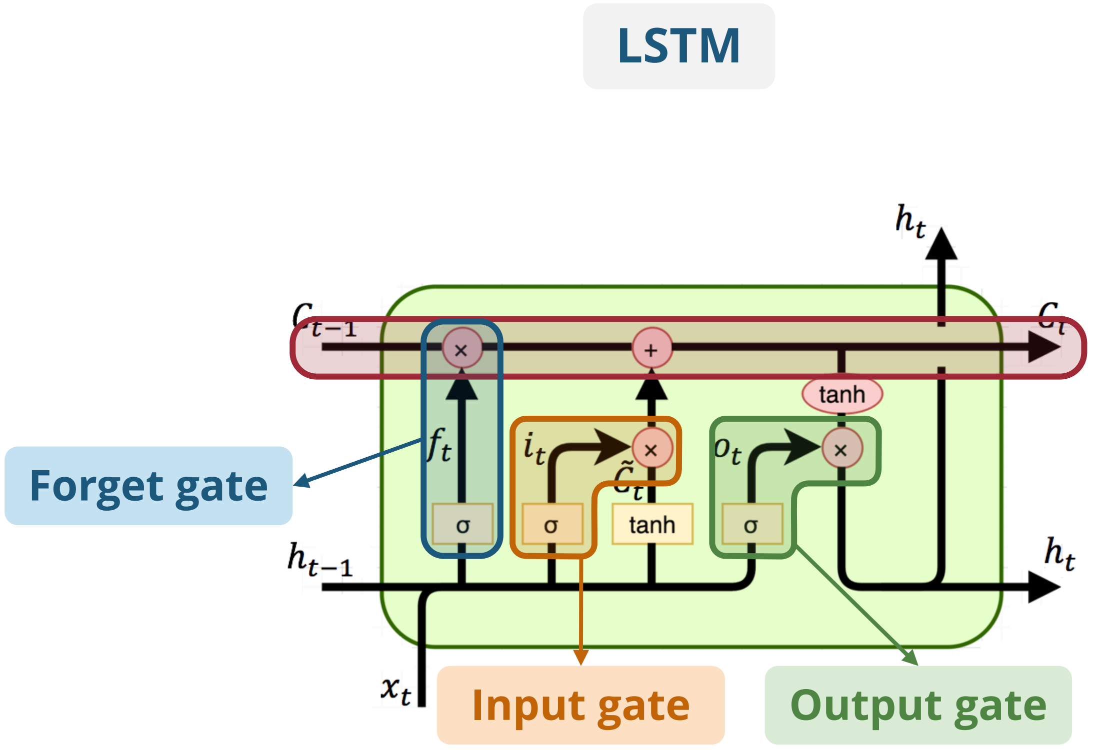
LSTMs does not completely solve vanishing gradients

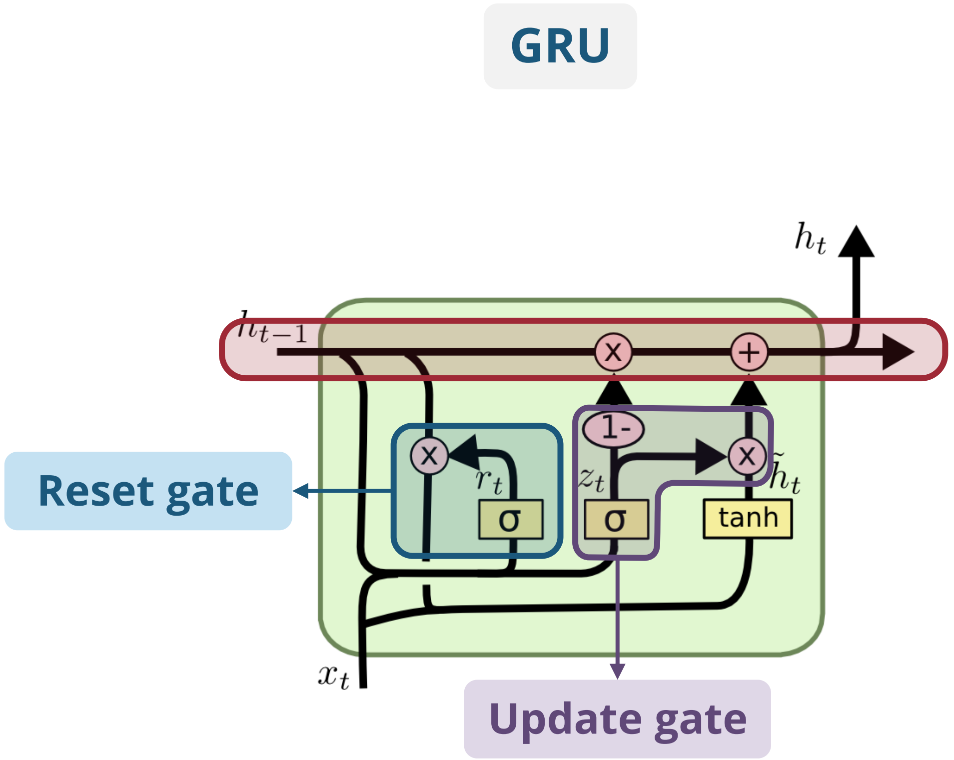
Gated Recurrent Units (GRUs)

- A **simplified** version of LSTM
- An LSTM consists of
  - **Forget** gate
  - **Input** gate
  - **Output** gate
- An GRU consists of
  - **Reset** gate
  - **Update** gate

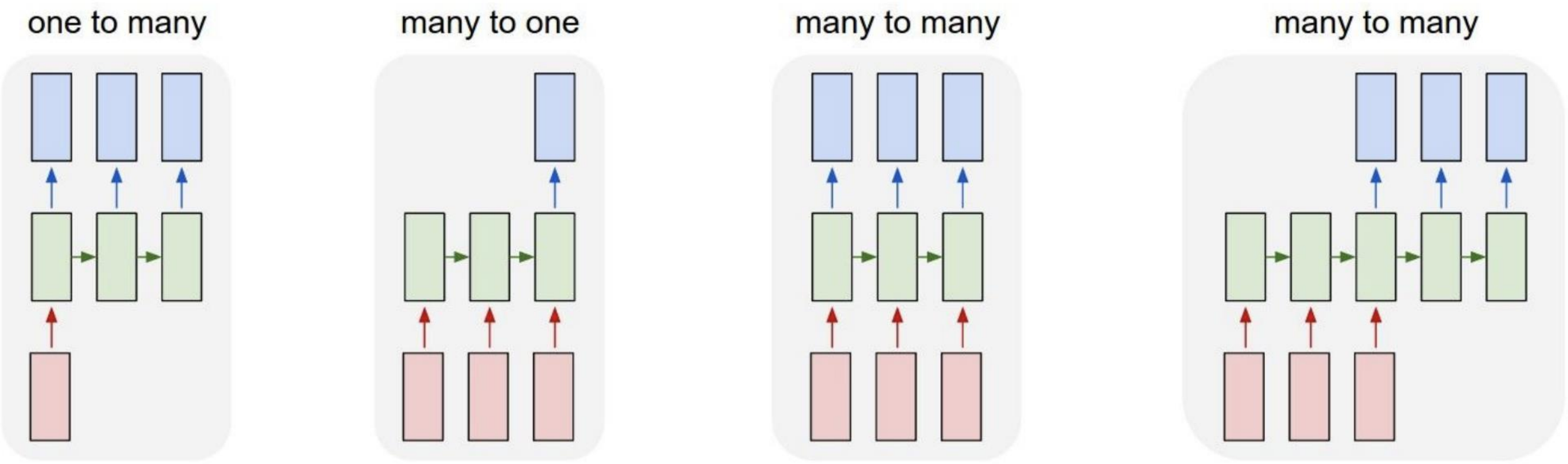


LSTMs vs GRUs





Different Types of Recurrent Neural Networks



Text generation
Music generation

Sentiment classification
Genre classification

Name entity recognition
Performance rendering

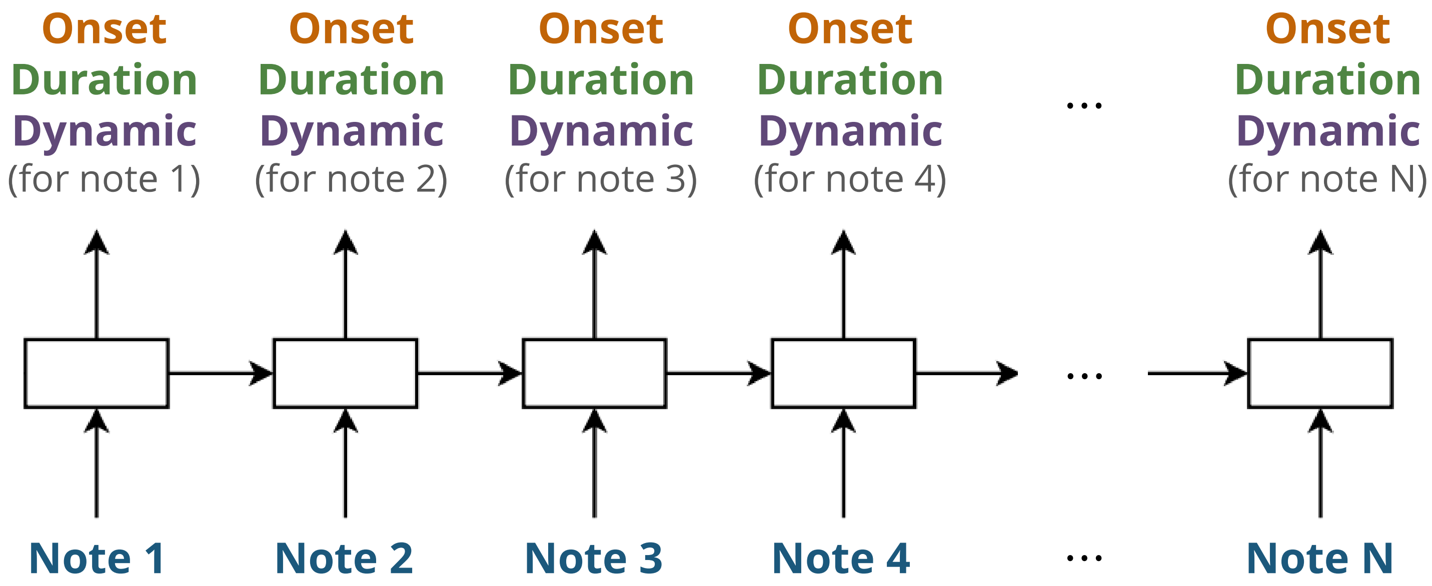
Machine translation
Music accompaniment
Style Transfer

(Source: CS231n)

cs231n.github.io/rnn/

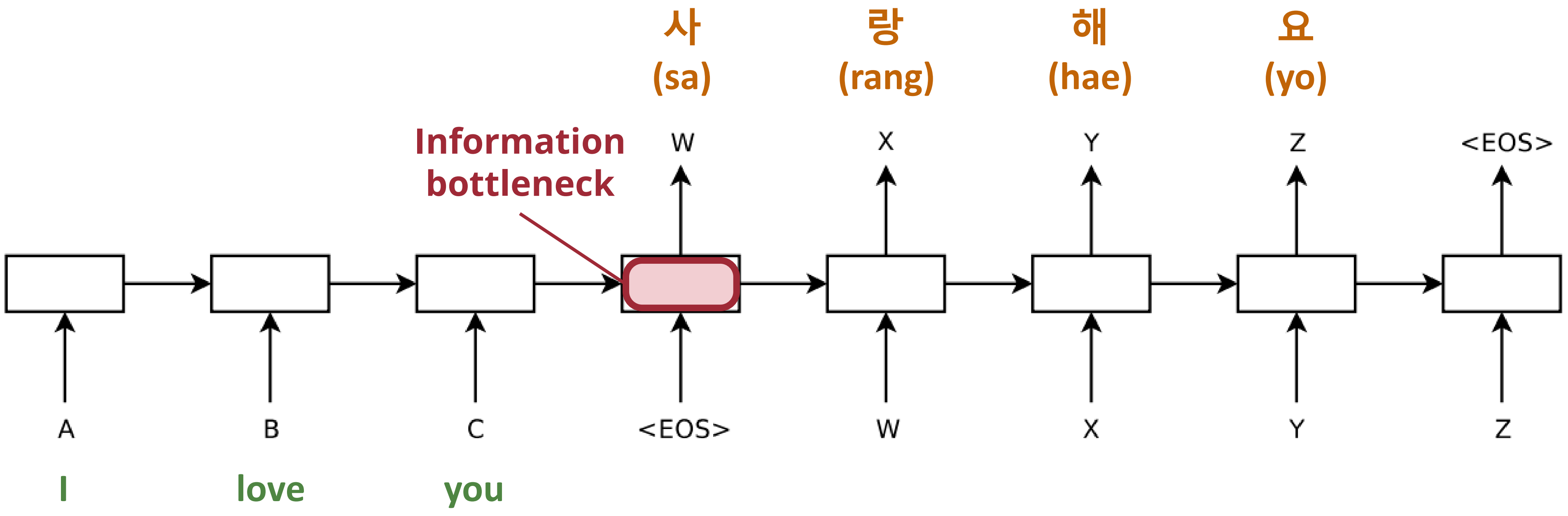
Many-to-Many RNNs

- Inputs and outputs are **aligned sequences**



Sequence-to-Sequence Model (Seq2seq)

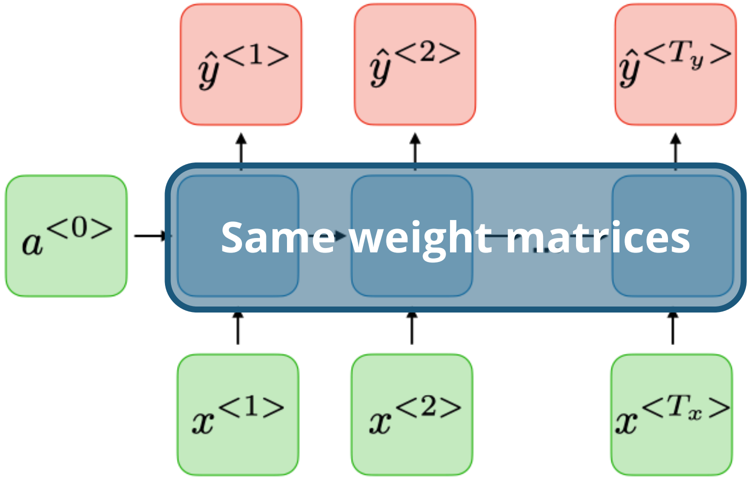
- Widely used for **machine translation**
- Inputs and outputs are **unaligned sequences**

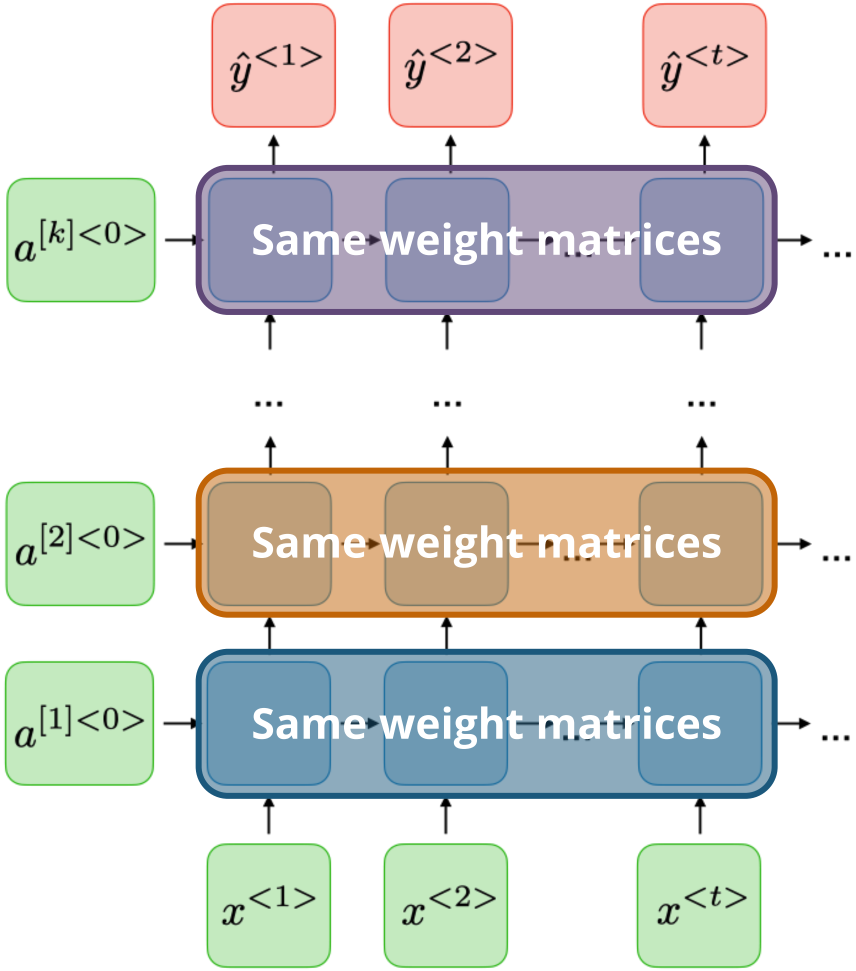


Ilya Sutskever, Oriol Vinyals, and Quoc V. Le, "Sequence to Sequence Learning with Neural Networks," *NeurIPS*, 2014.

Variants of RNNs

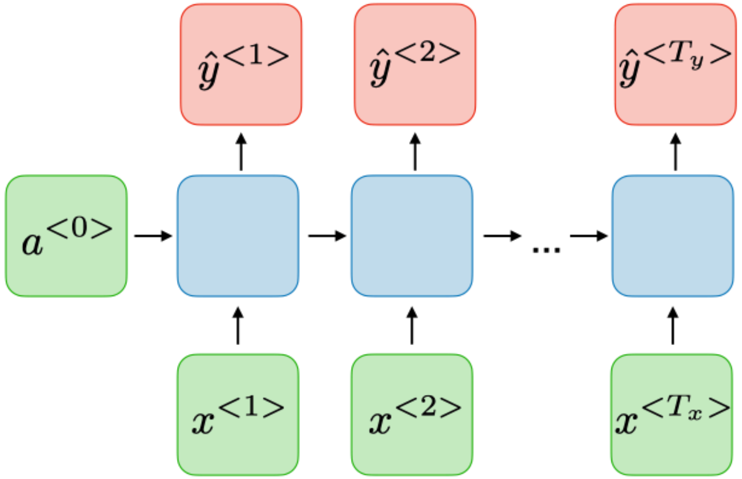
Deep Recurrent Neural Networks



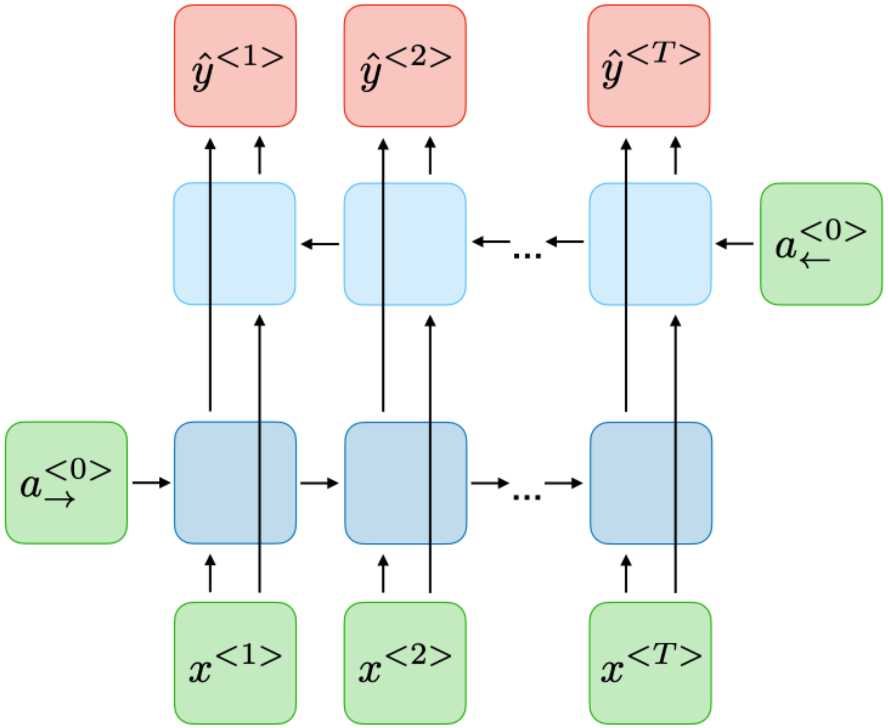


stanford.edu/~shervine/teaching/cs-230/cheatsheet-recurrent-neural-networks

Bidirectional RNNs



Access to only past information



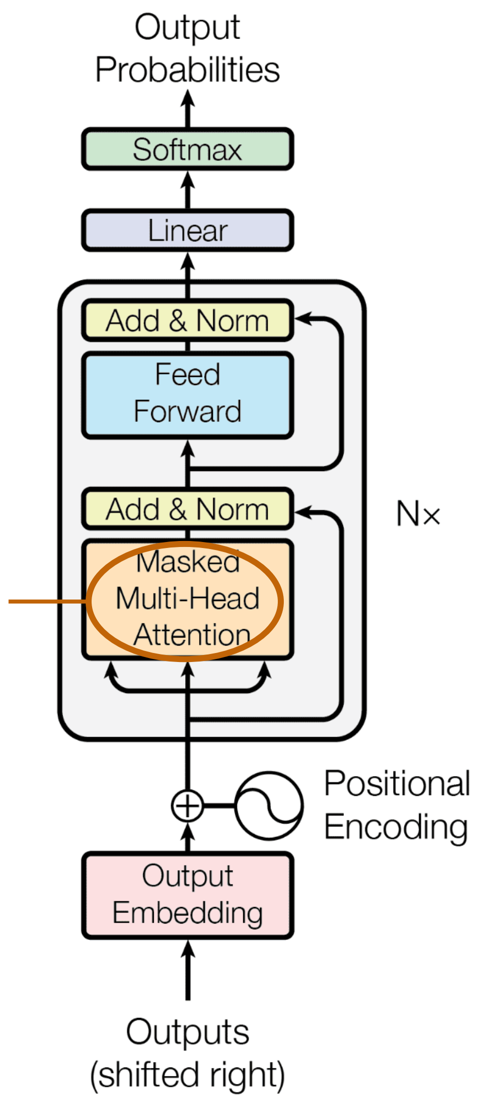
Access to past and future information

Transformers

What is a Transformer?

- A type of neural network that use the **self-attention mechanism**

Self-attention

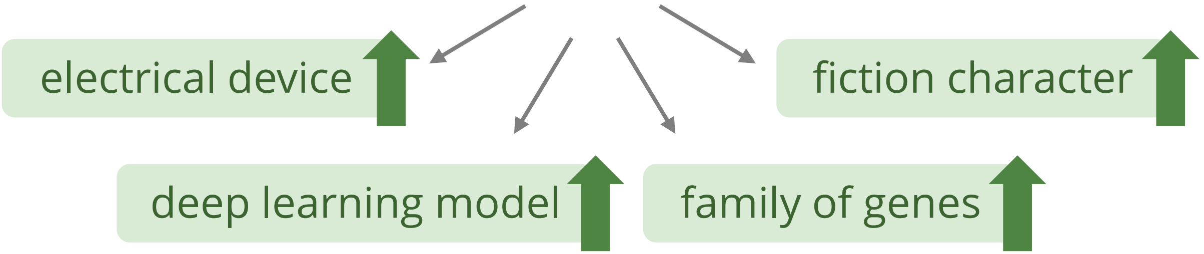


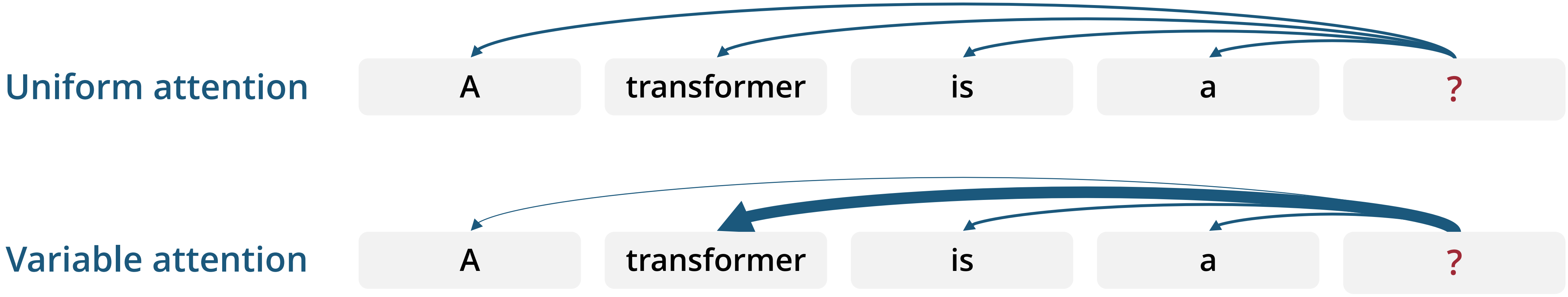
(Source: Vaswani et al., 2017; adapted)

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin, "Attention Is All You Need," *NeurIPS*, 2017.

Self-attention Mechanism

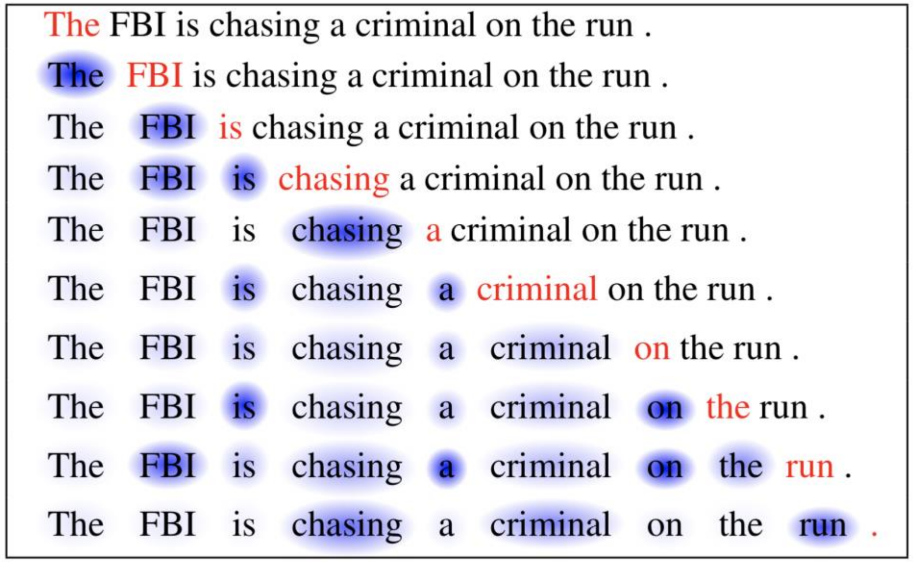
A transformer is a ________



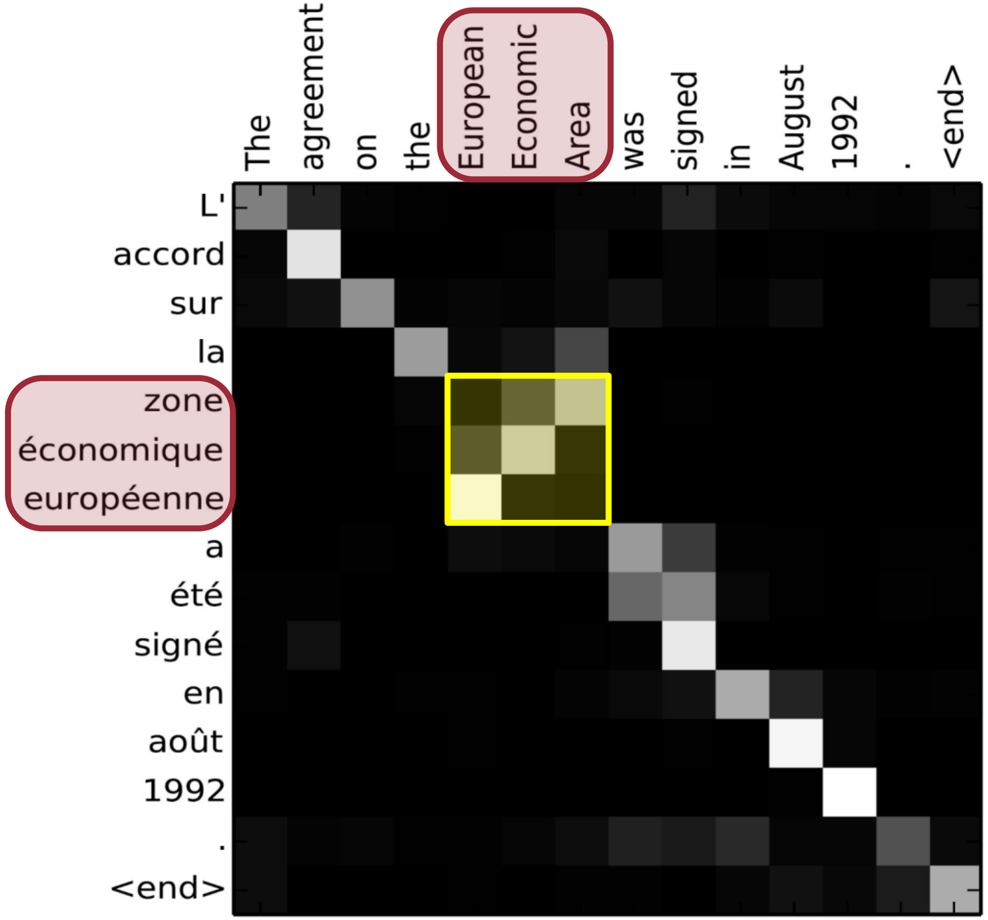


Transformers learn what to attend to from big data!

Why Attention Mechanism?



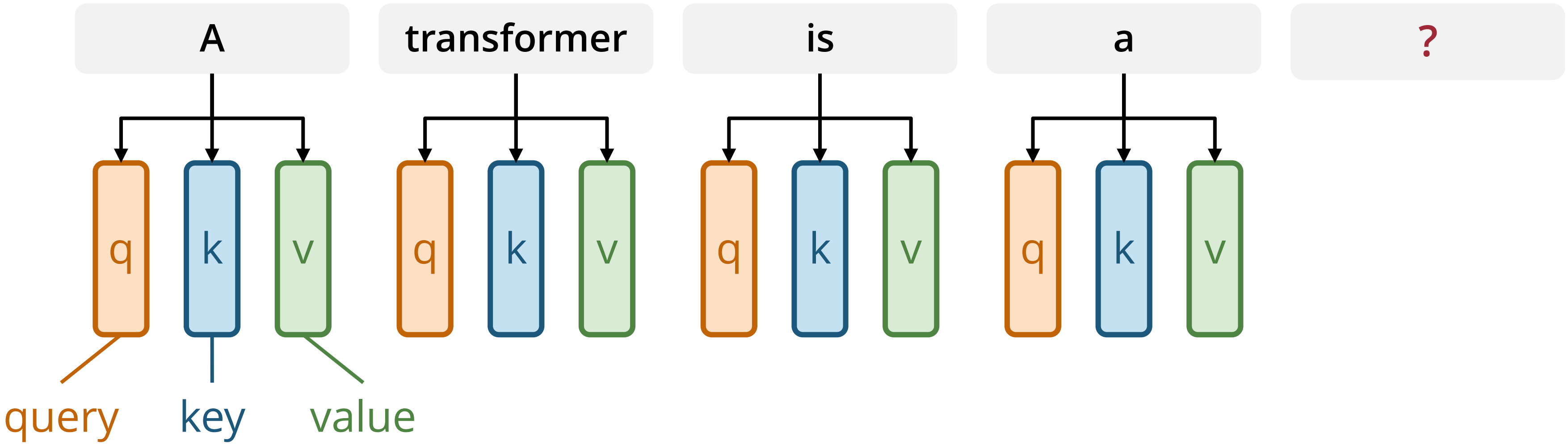
(Source: Cheng et al., 2016)



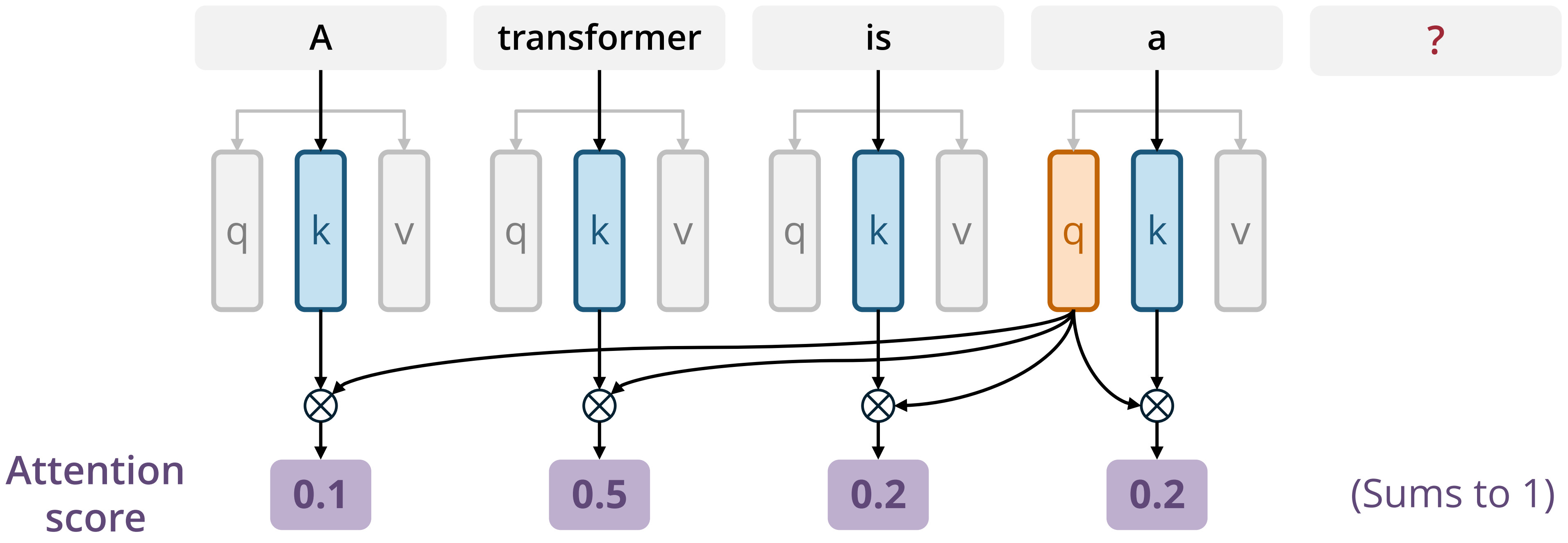
(Source: Bahdanau et al., 2015)

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio, "Neural Machine Translation by Jointly Learning to Align and Translate," *ICLR*, 2015.
Jianpeng Cheng, Li Dong, and Mirella Lapata, "Long Short-Term Memory-Networks for Machine Reading," *EMNLP*, 2016.

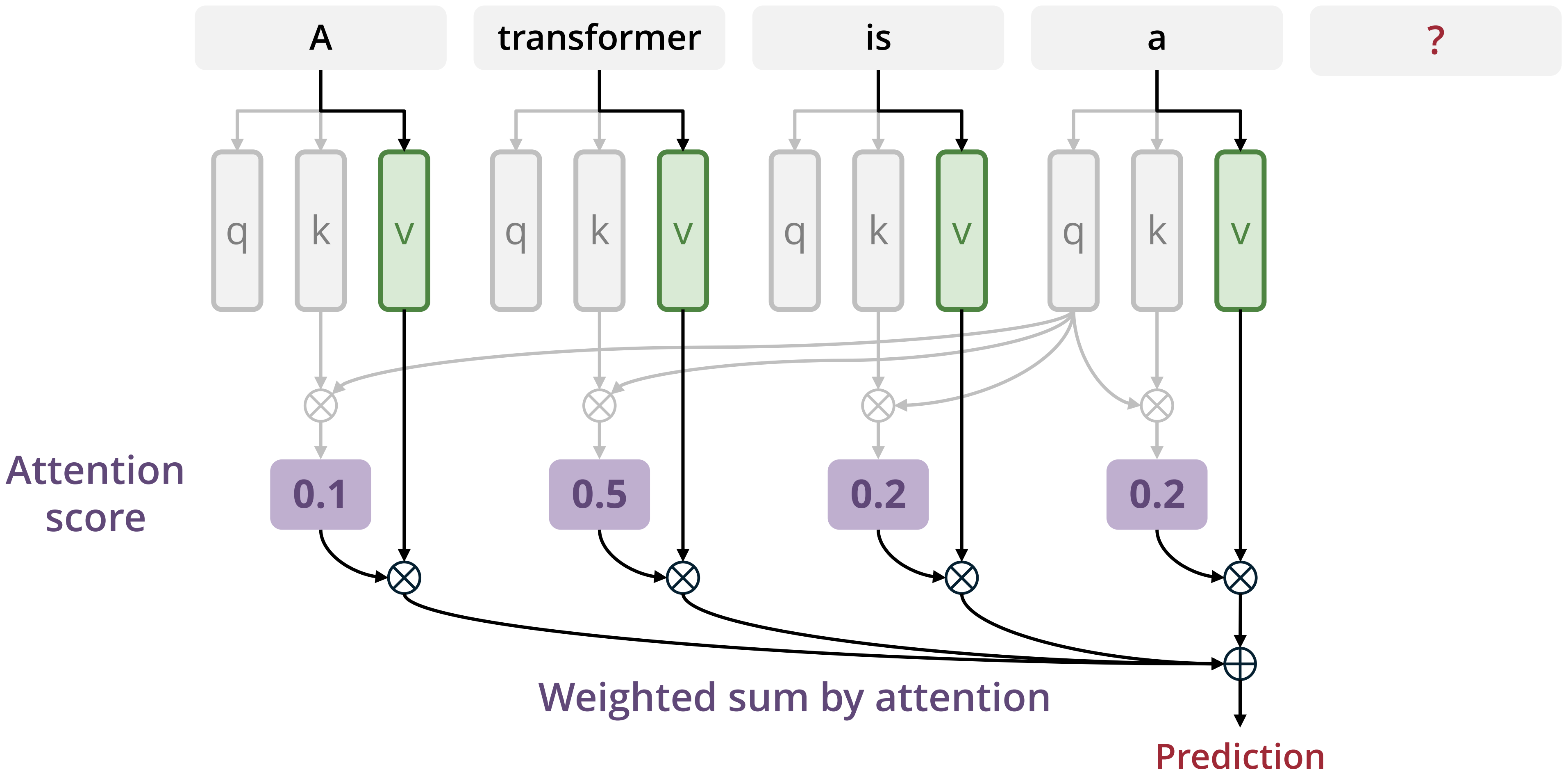
Demystifying Transformers



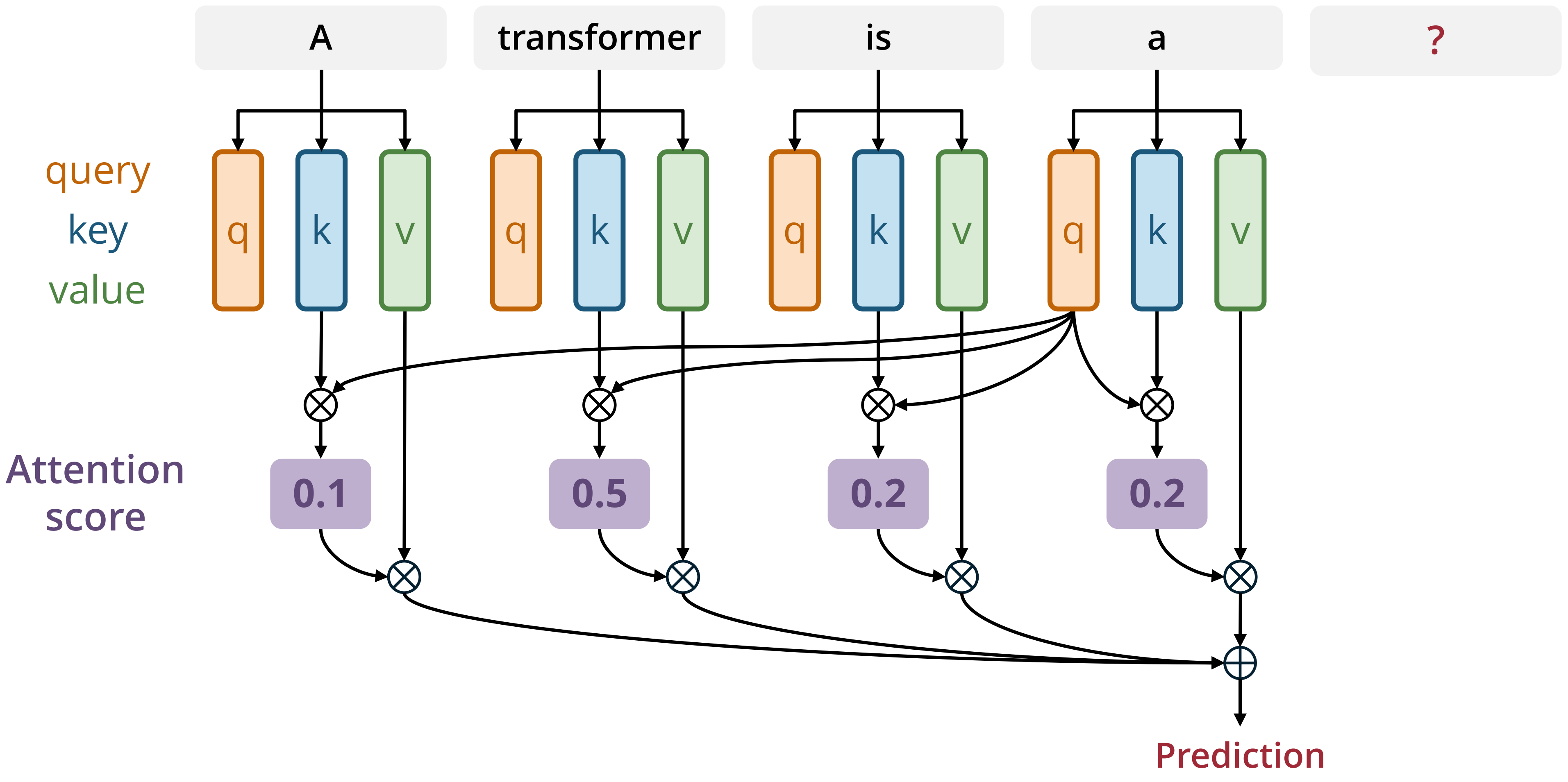
Demystifying Transformers



Demystifying Transformers

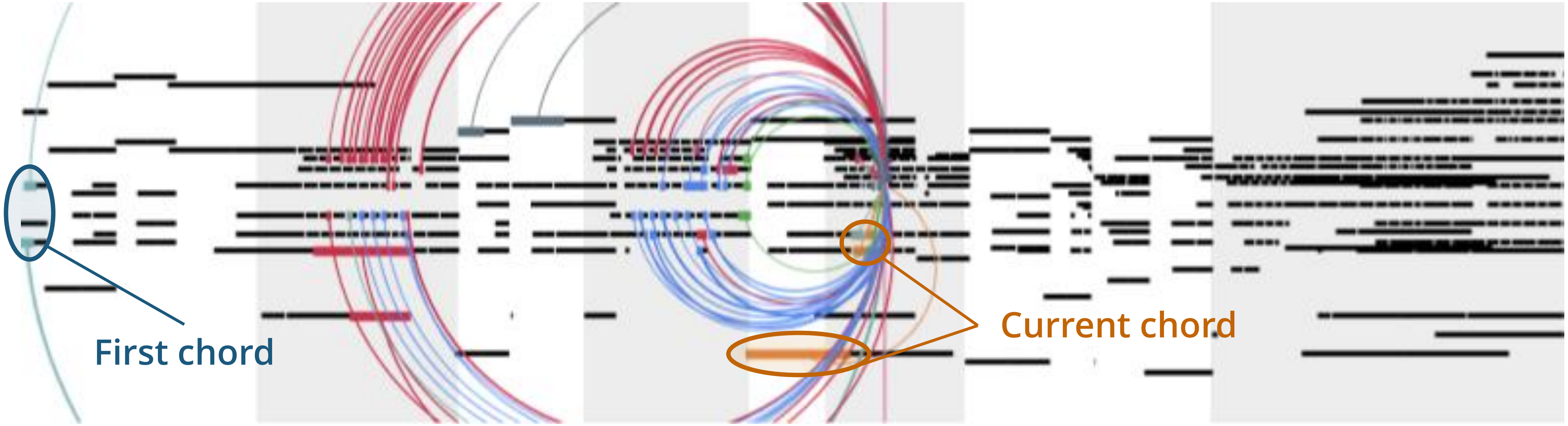


Demystifying Transformers



What does a Transformer Learn?

(Each color represents an attention head)

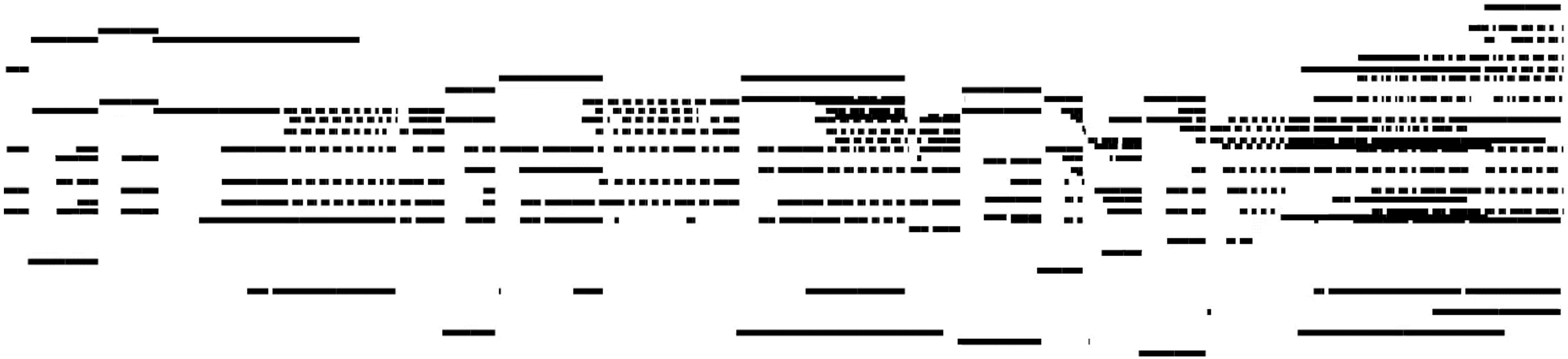


(Source: Huang et al., 2018)

Cheng-Zhi Anna Huang, Ashish Vaswani, Jakob Uszkoreit, Noam Shazeer, Ian Simon, Curtis Hawthorne, Andrew M. Dai, Matthew D. Hoffman, Monica Dinculescu, and Douglas Eck, "Music Transformer: Generating Music with Long-Term Structure," *Magenta Blog*, December 13, 2018.

What does a Transformer Learn?

(Each color represents an attention head)

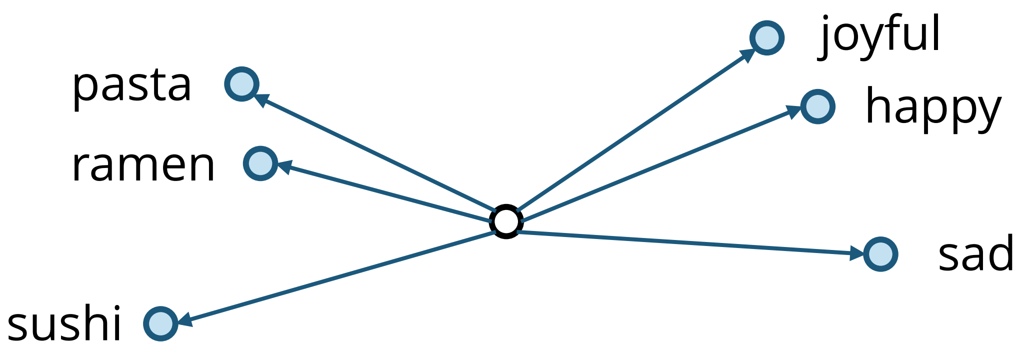


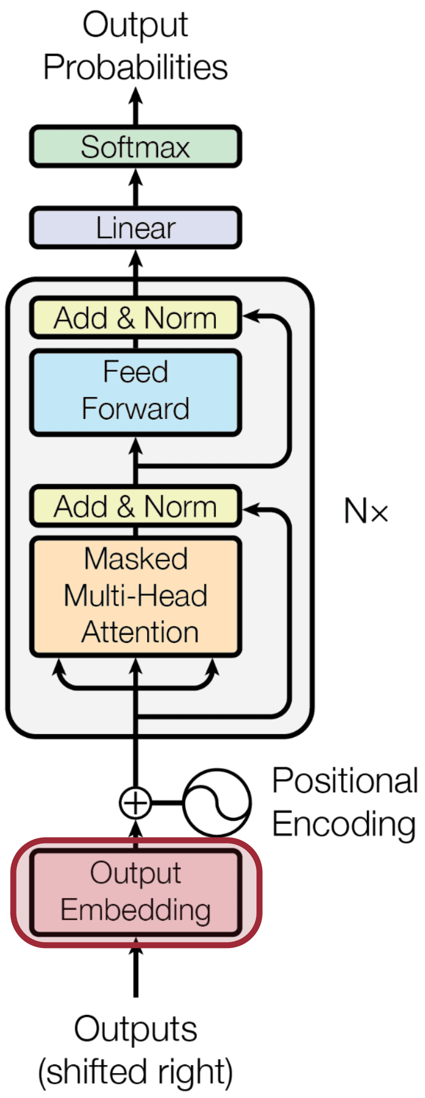
(Source: Huang et al., 2018)

Cheng-Zhi Anna Huang, Ashish Vaswani, Jakob Uszkoreit, Noam Shazeer, Ian Simon, Curtis Hawthorne, Andrew M. Dai, Matthew D. Hoffman, Monica Dinculescu, and Douglas Eck, "Music Transformer: Generating Music with Long-Term Structure," *Magenta Blog*, December 13, 2018.

Word Embedding

- **Goal**: Learn to represent words as vectors
- **Intuition**: Synonyms should have close embeddings
- Antonyms should be far apart?
  - Not quite, antonyms usually fall in the same "topic"
  - For example, happy and sad are antonyms, but they are both emotions



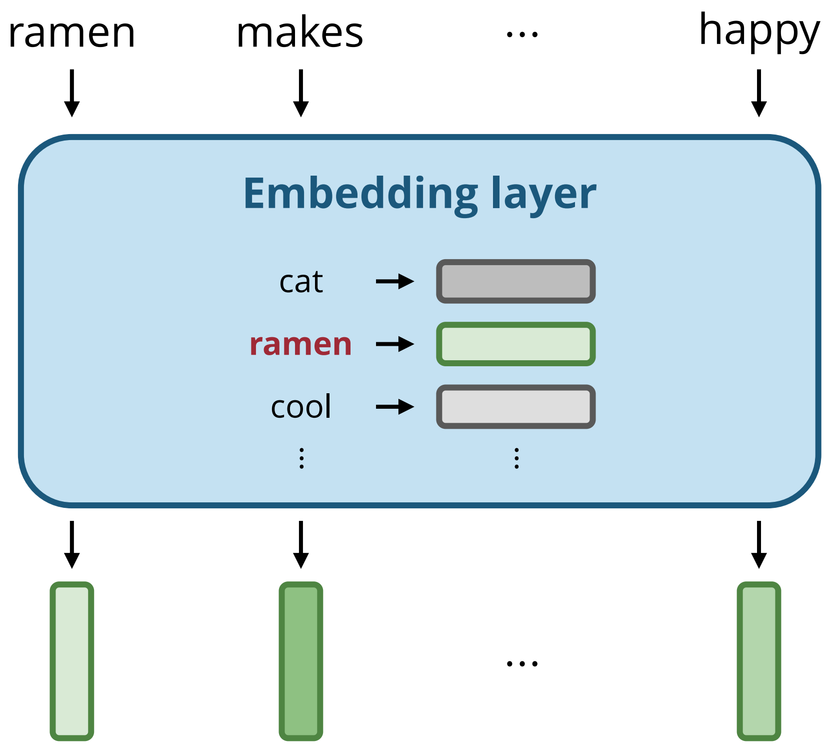


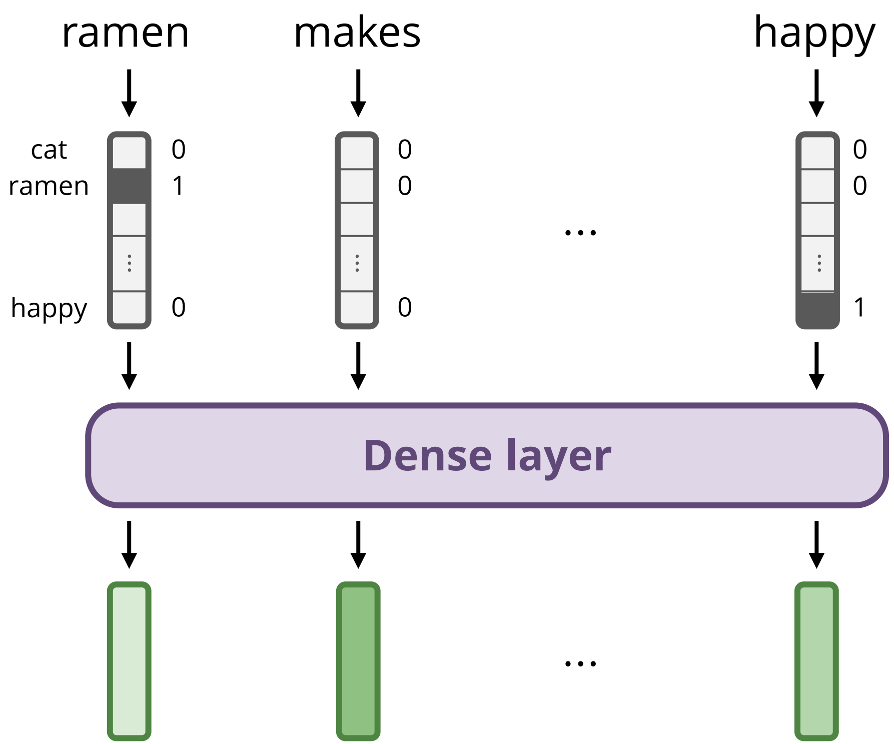
(Source: Vaswani et al., 2017; adapted)

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin, "Attention Is All You Need," *NeurIPS*, 2017.

Word Embedding

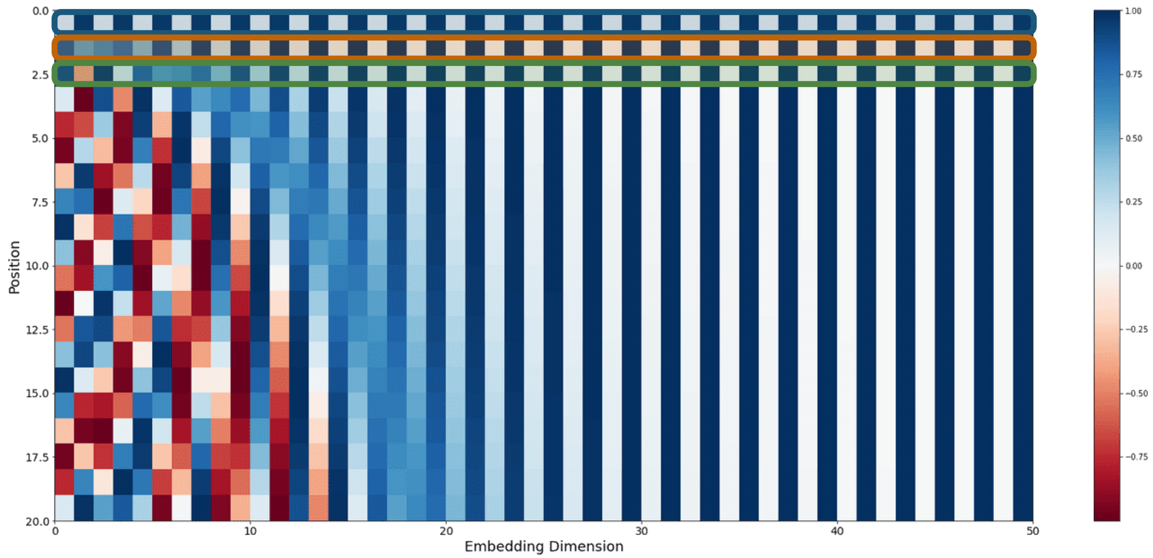
- A **word embedding layer** is functionally equivalent to **one-hot encoded words** followed by a **dense layer** → **But way faster!**

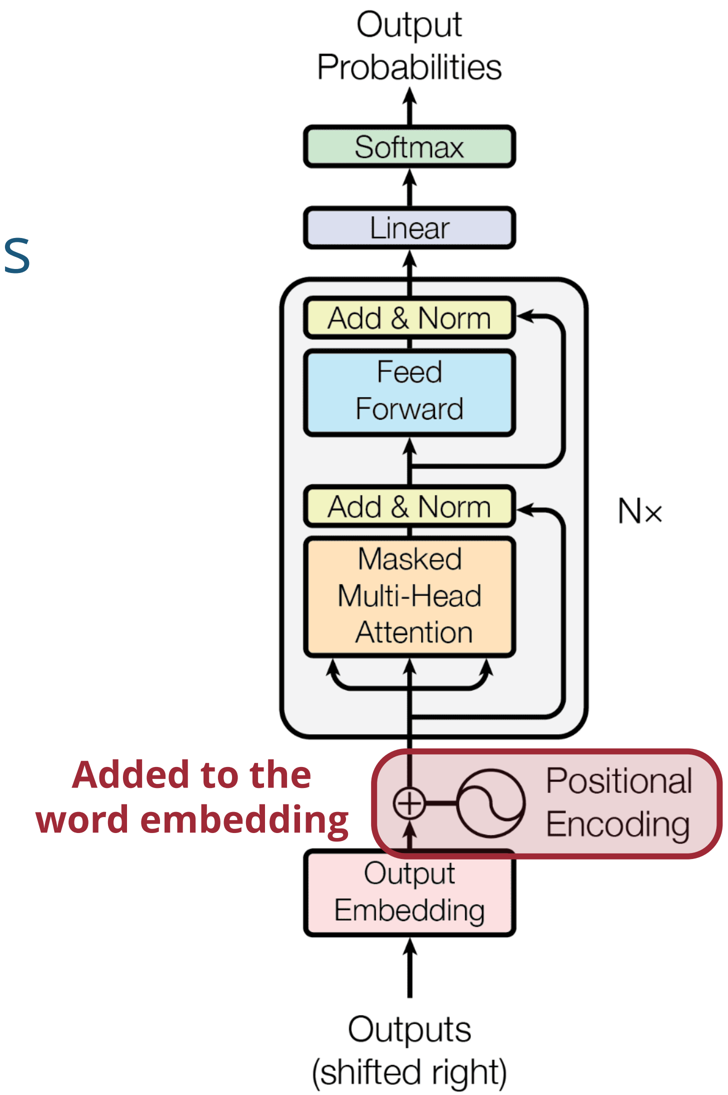




Positional Encoding

- **Intuition**: A word could have different meanings at different positions
- Provides **positional information** to the model



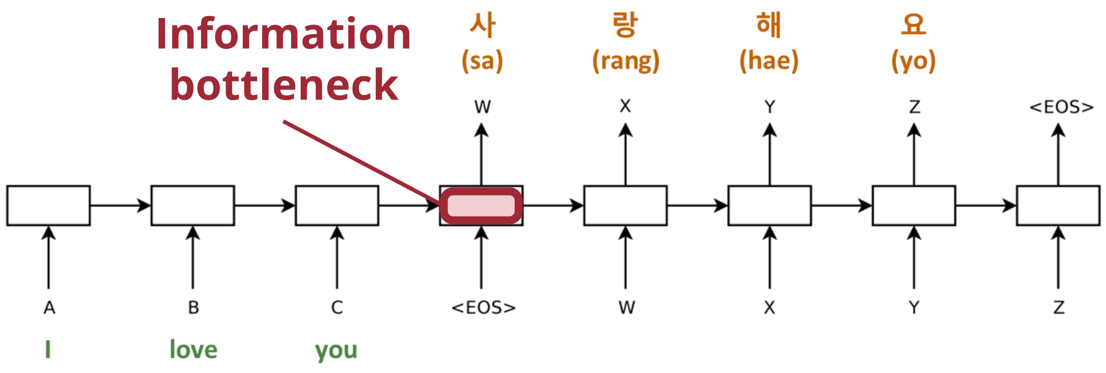


(Source: Vaswani et al., 2017; adapted)

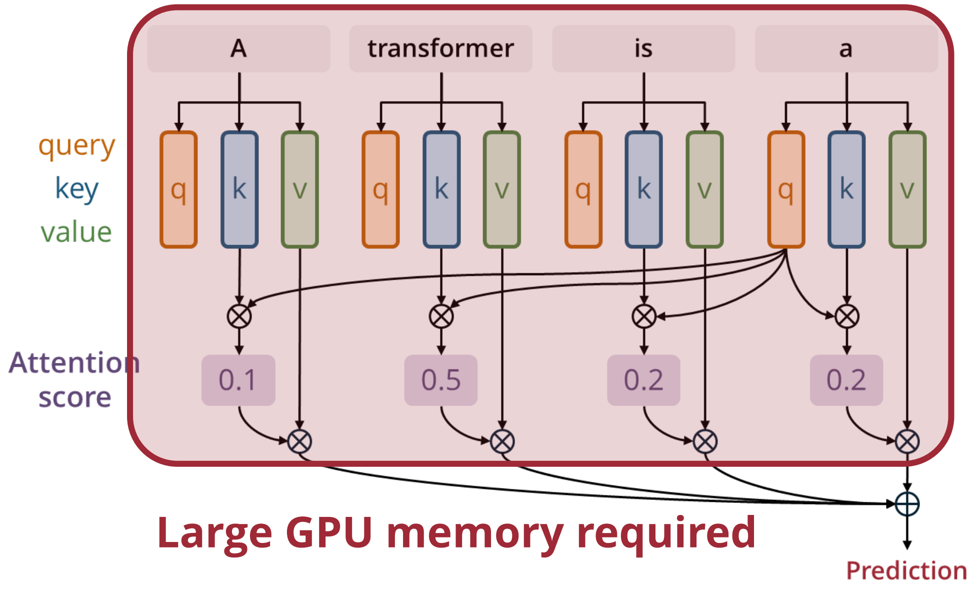
Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin, "Attention Is All You Need," *NeurIPS*, 2017.
erdem.pl/2021/05/understanding-positional-encoding-in-transformers

Seq2seq vs Transformers

Seq2seq

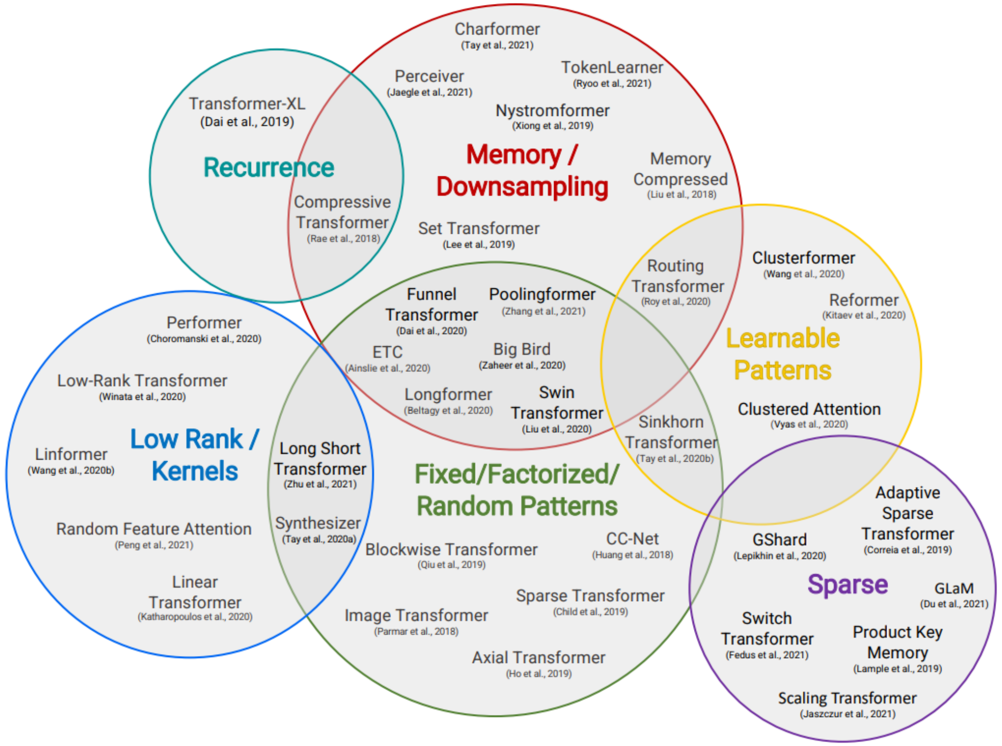


Transformers



Efficient Transformers

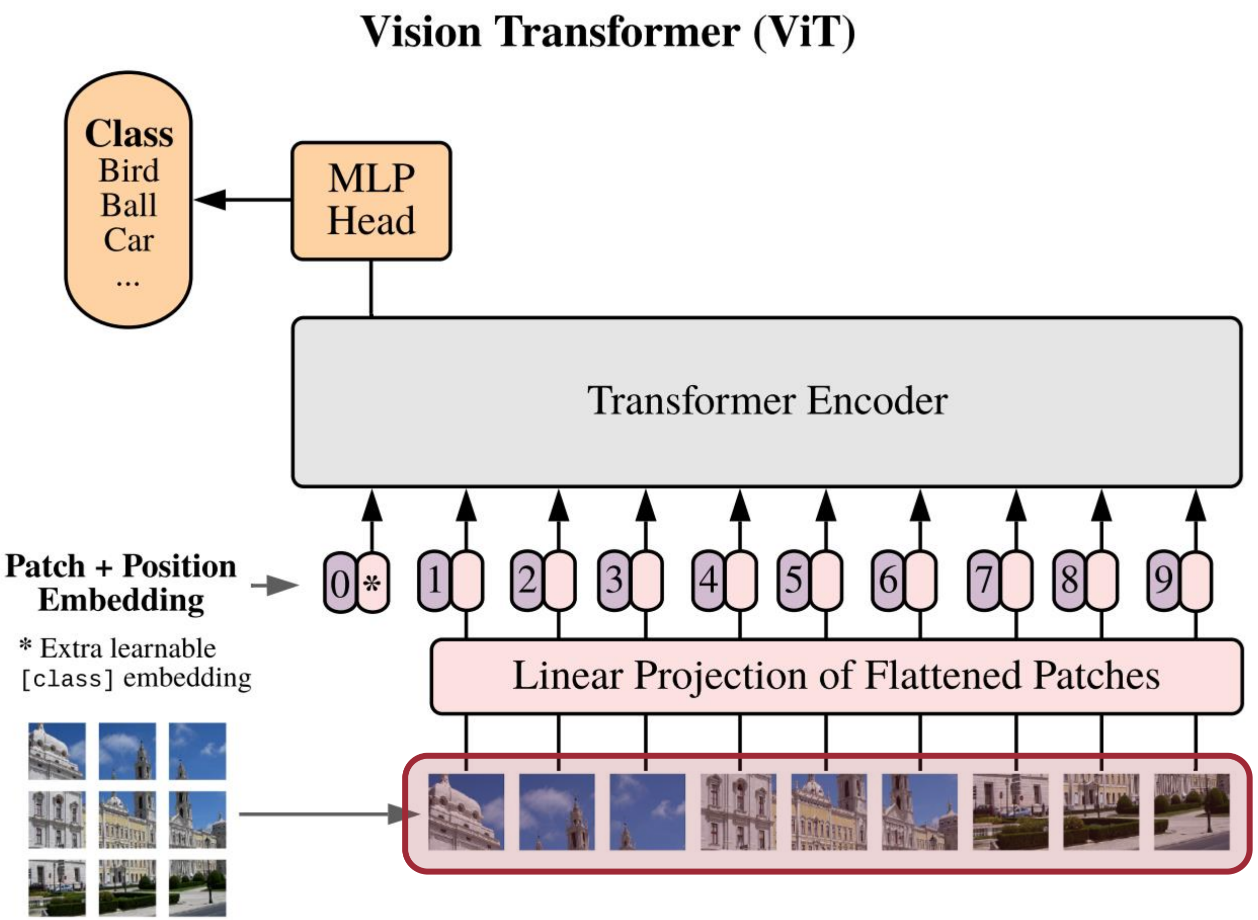
- The **memory requirement for self-attention** grows **quadratically**!
- There are many efficient transformer variants
  - Transformer-XL
  - Linear Transformer
  - Performer
  - Longformer
  - Reformer
  - Swin Transformer
  - *… just to name a few*



(Source: Tay et al., 2022)

Yi Tay, Mostafa Dehghani, Dara Bahri, and Donald Metzler, "Efficient Transformers: A Survey," *arXiv preprint arXiv:2009:06732*, 2022.

Vision Transformer (ViT)

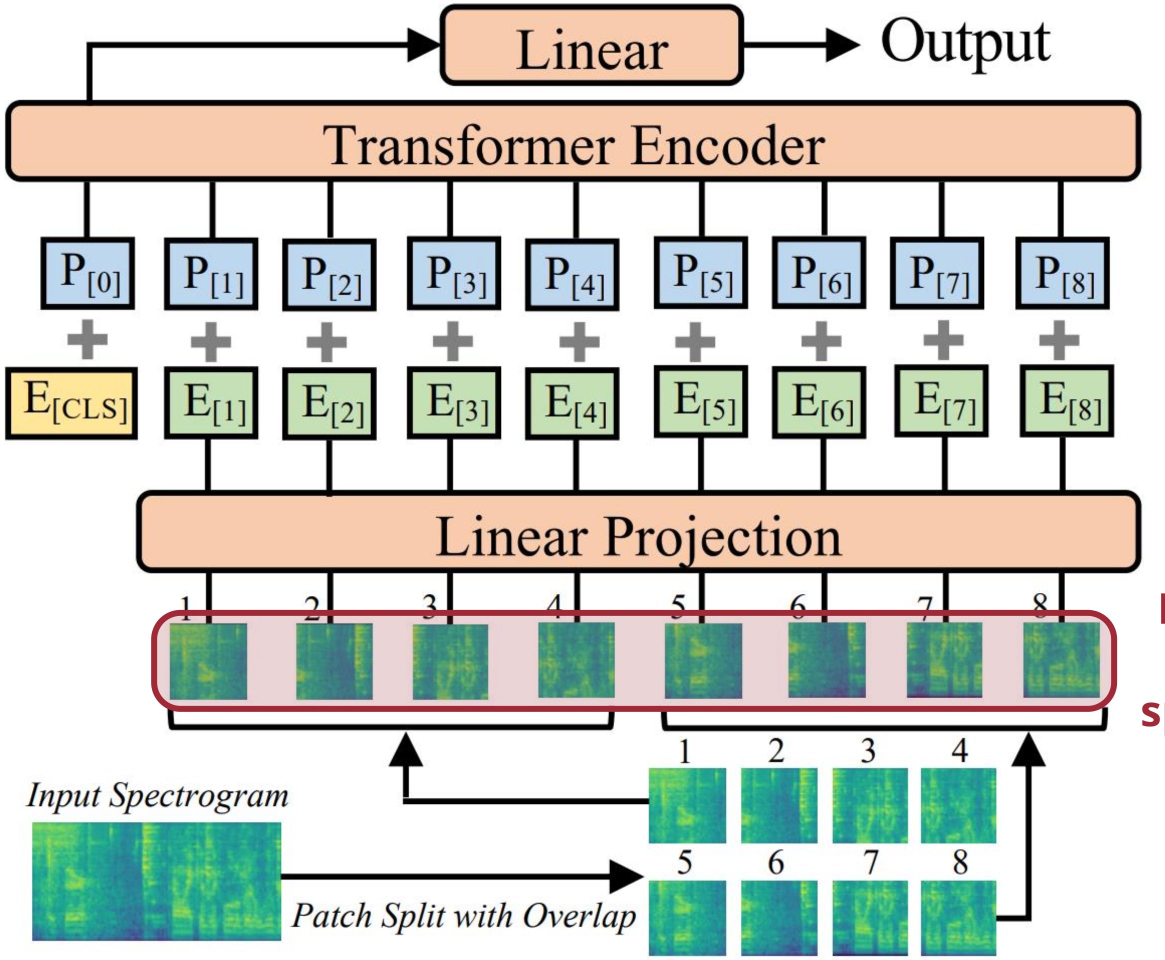


Representing an image as a sequence of patches

(Source: Dosovitskiy et al., 2021)

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," *ICLR*, 2021.

Audio Spectrogram Transformer (AST)



Representing audio as a sequence of spectrogram patches

(Source: Gong et al., 2021)

Yuan Gong, Yu-An Chung, and James Glass, "AST: Audio Spectrogram Transformer," *INTERSPEECH*, 2021.