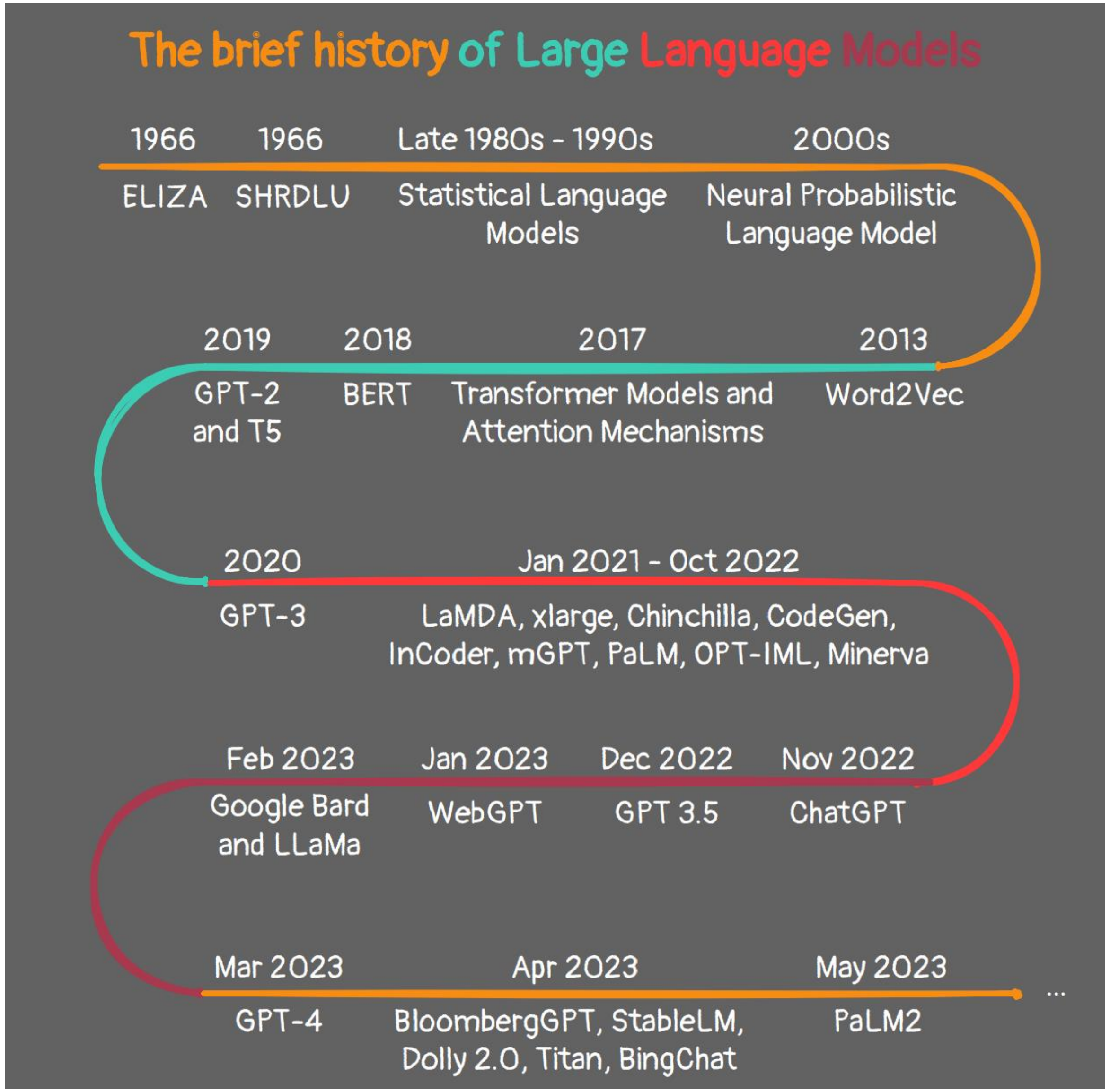
transformers and beyond

GESIS Fall Seminar 2023

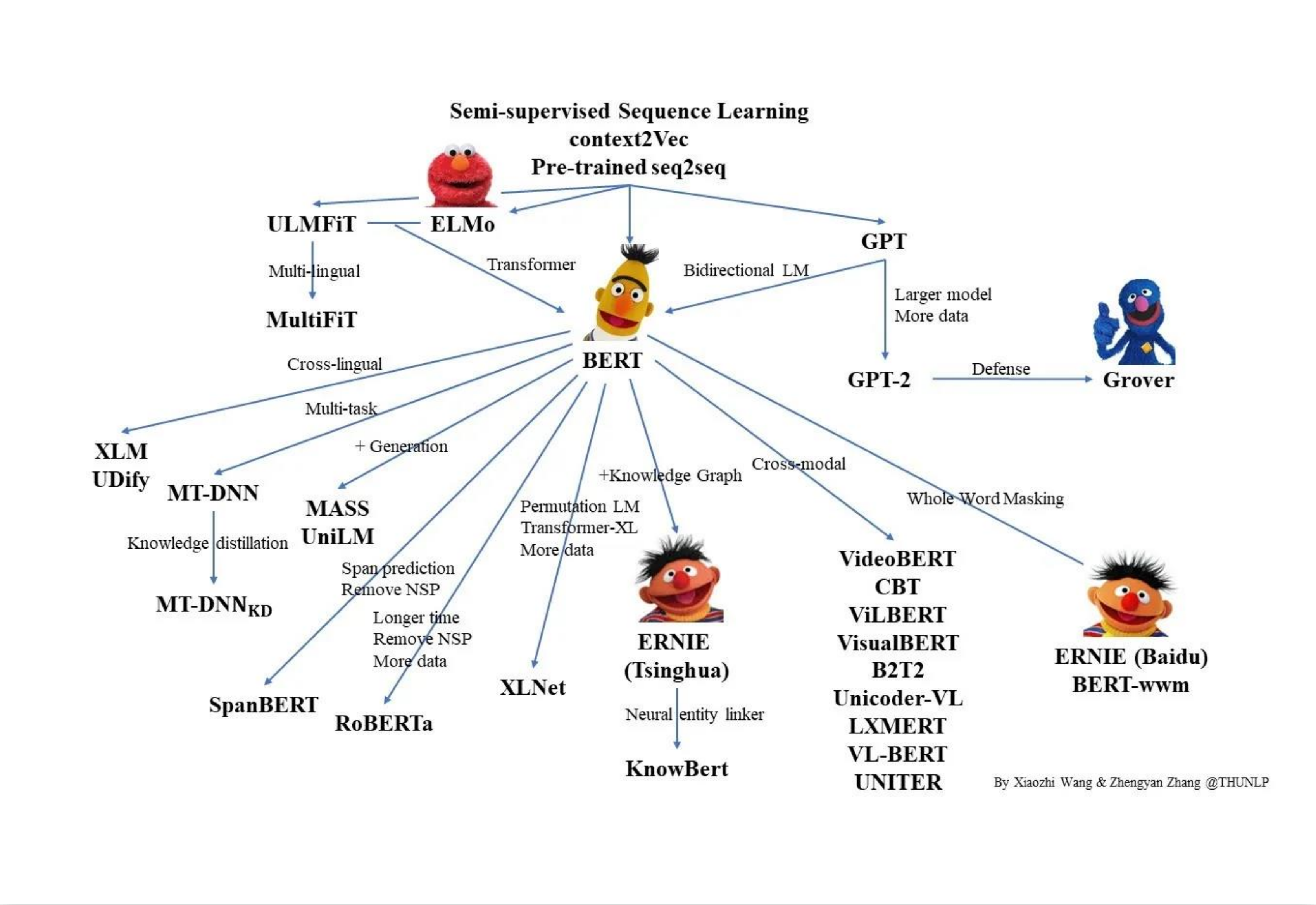
“From Embeddings to Transformers: Advanced Text Analysis in Python”

[day 3, afternoon: attention]

some history



[The brief history of Large Language Models: A Journey from ELIZA to GPT-4 and Google Bard | by Armin Norouzi, Ph.D | Level Up Coding (gitconnected.com)]

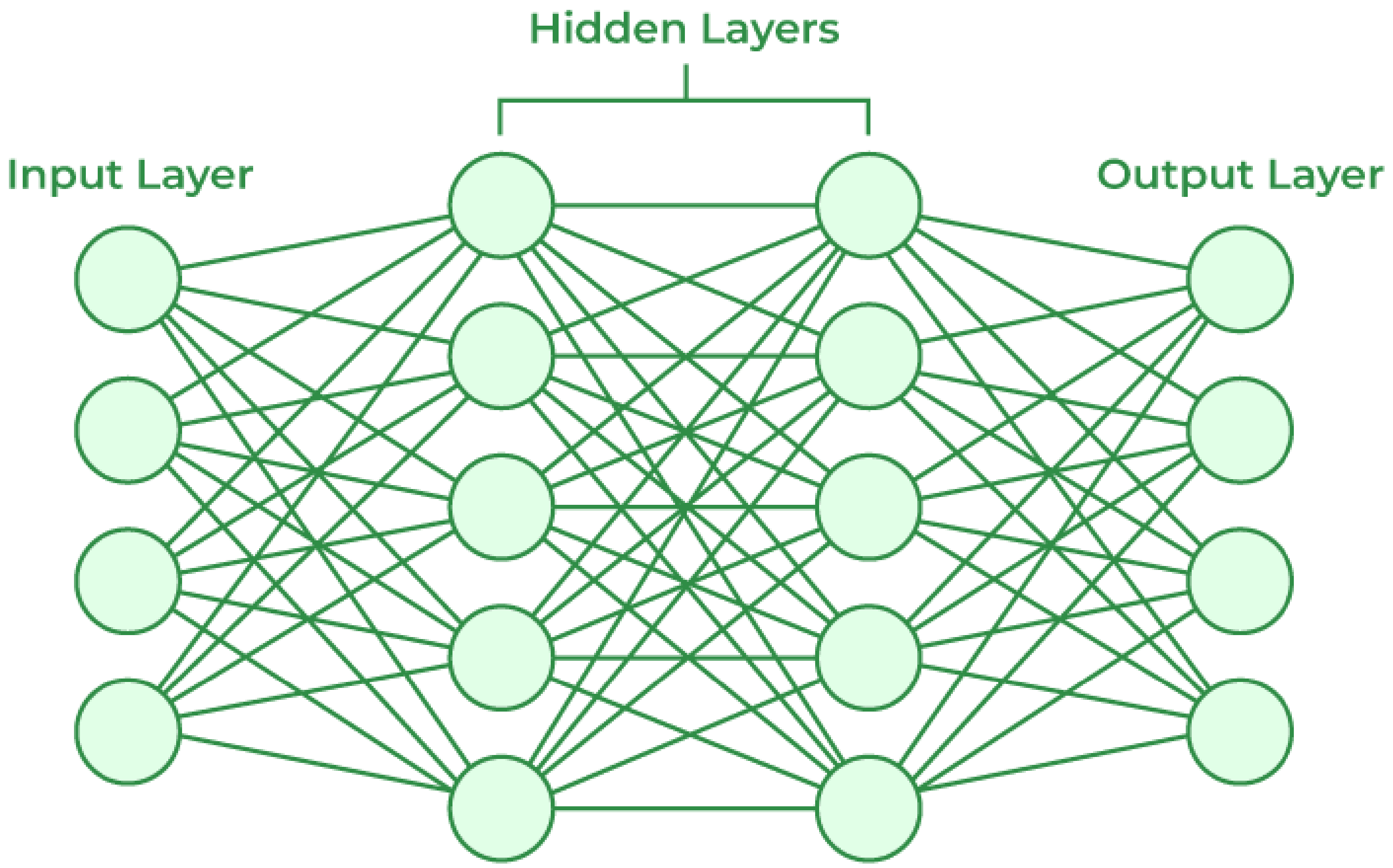


ELMo, BERT, & crew aka GPT

[10 Things You Need to Know About BERT and the Transformer Architecture That Are Reshaping the AI Landscape (neptune.ai)]

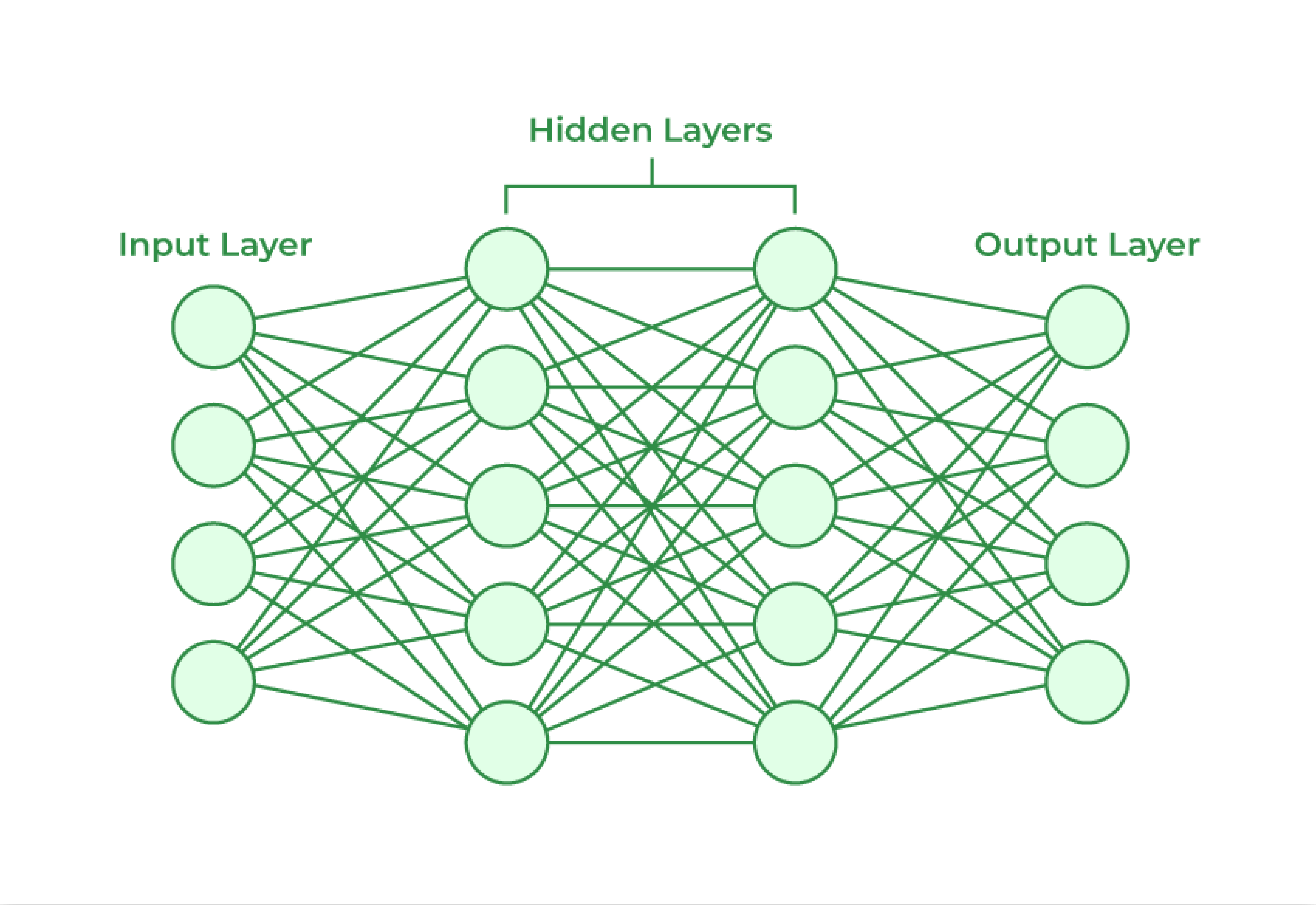
neural networks

[Neural Network In 5 Minutes | What Is A Neural Network? | How Neural Networks Work | Simplilearn - YouTube]



neural networks

[Artificial Neural Networks and its Applications - GeeksforGeeks]



does this look familiar?

don’t think too far!

regression?

[Lesson 3: Neural Network is nothing but a Linear Regression | by Md. Asifur Rahman | Medium]

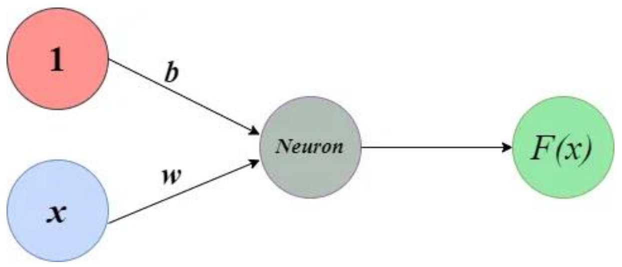
linear regression!

$$y = \beta_0 + \beta_1 X + \varepsilon$$

$$y = mx + c$$

y=mx+c; ***x*** is **feature vector**, ***c*** is **bias**, ***y*** is **output** or **dependent variable**, **m** is **weight vector**.

single neural network path!



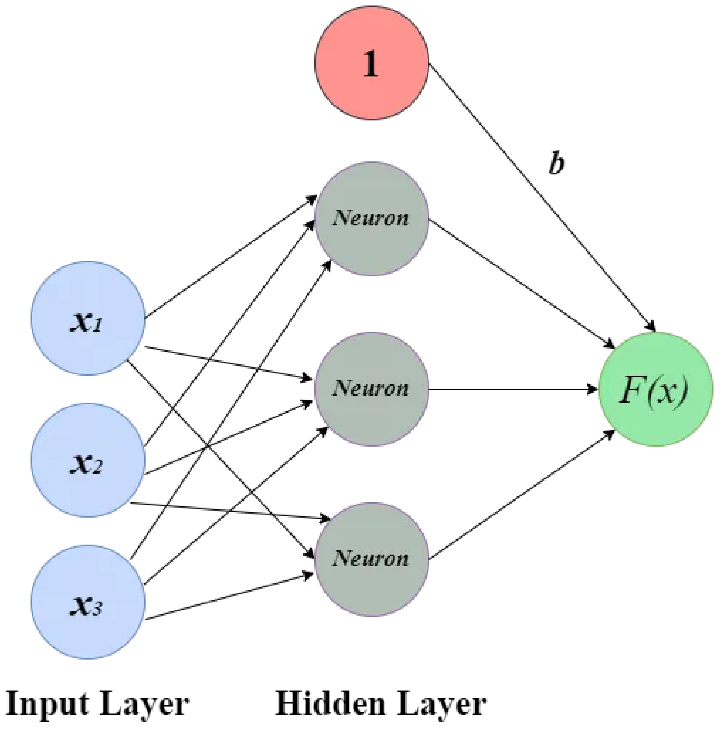
$$F(x) = w^T x + b$$

[Lesson 3: Neural Network is nothing but a Linear Regression | by Md. Asifur Rahman | Medium]

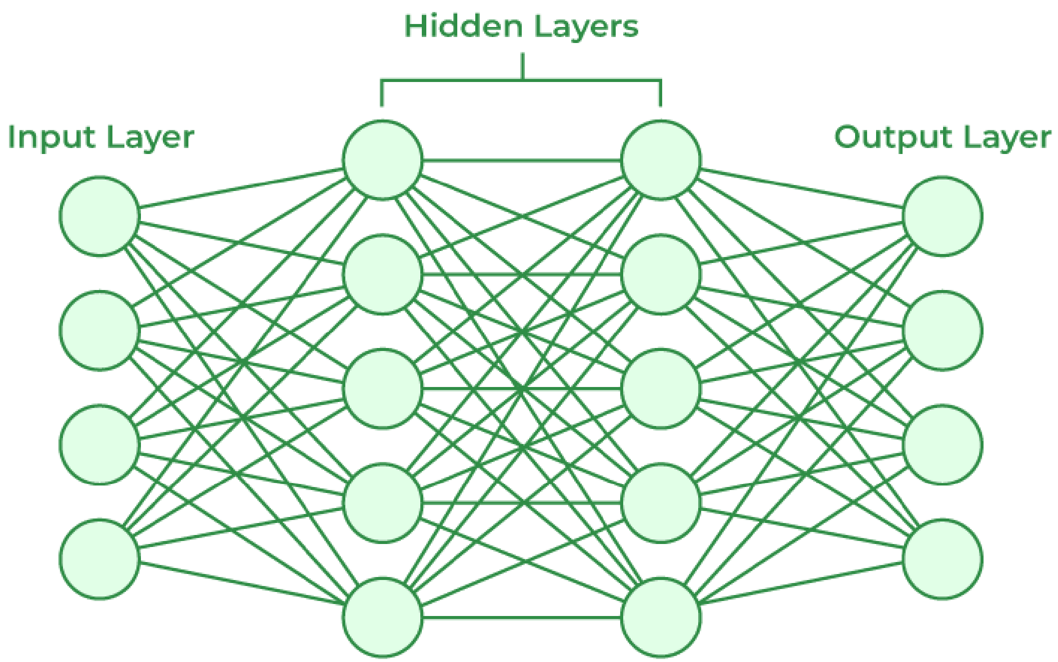
multiple linear regression!   simple perceptron!

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ \ldots + \beta_i X_i$$

Y : Dependent variable
β_0 : Intercept
β_i : Slope for X_i
X = Independent variable



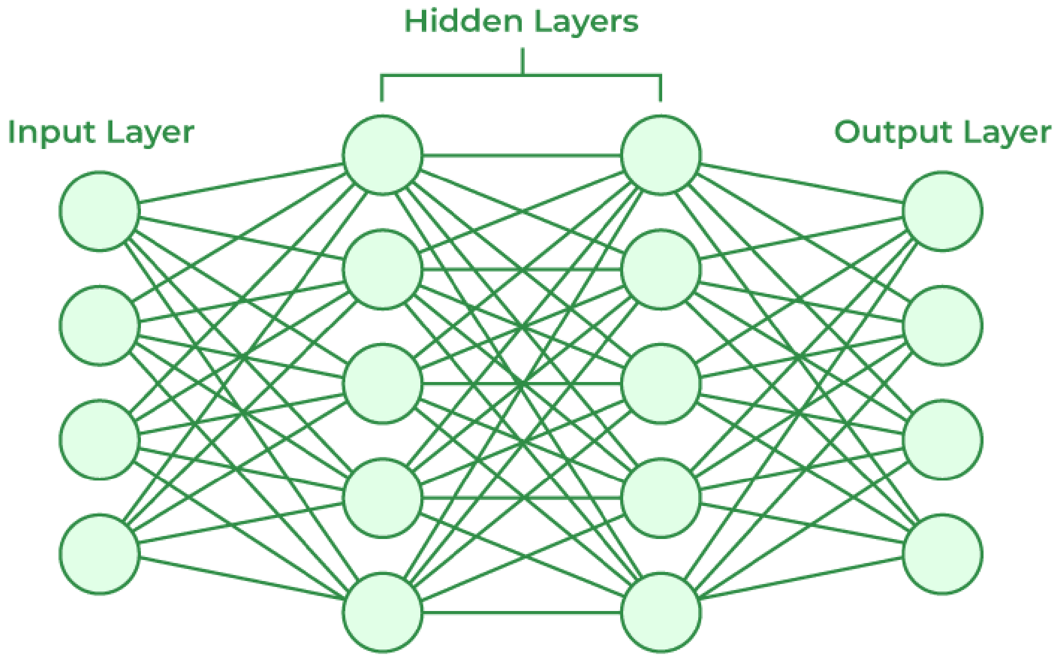
[[Lesson 3: Neural Network is nothing but a Linear Regression | by Md. Asifur Rahman | Medium]] adding additional layers ➡ neural network!



but what about linearity and non-linearity?

linear regression = linear.

neural networks = solve complex non-linear problems.

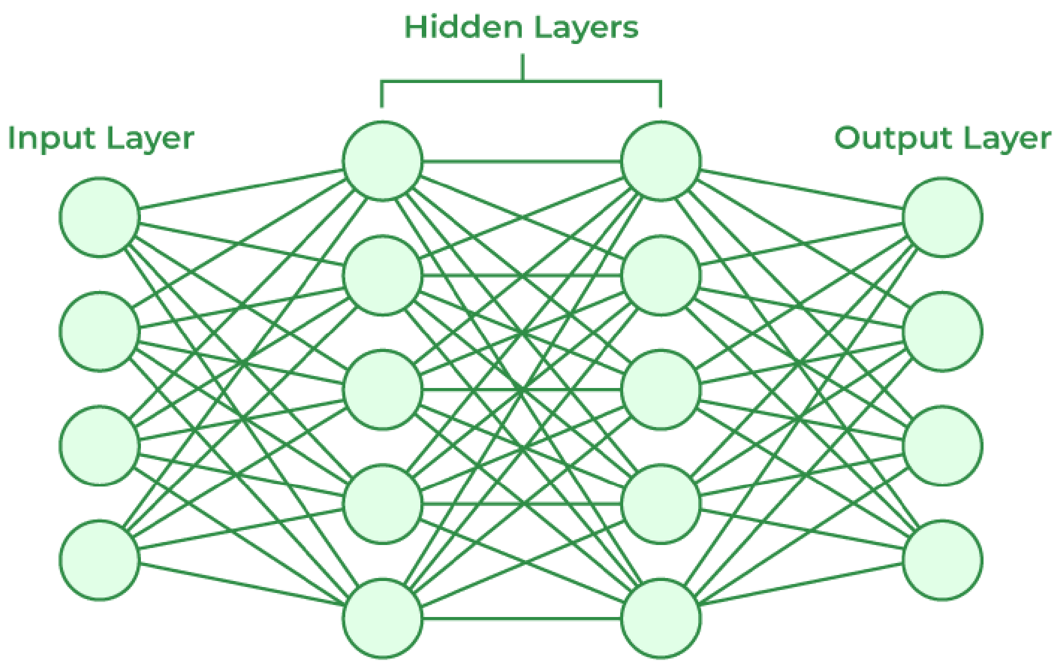


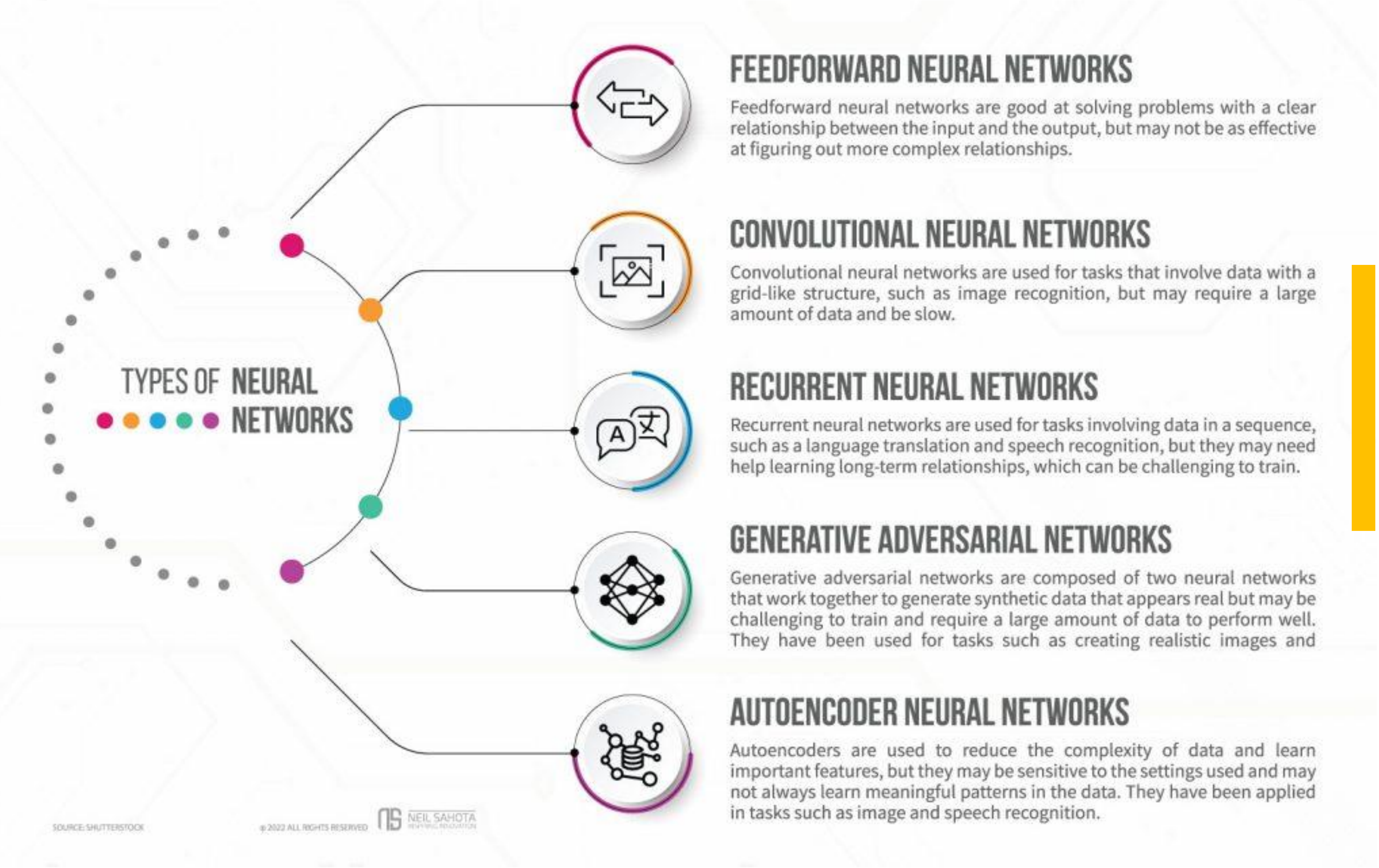
remember the activation function

If we exchange a non-linear activation function

with a linear activation function,

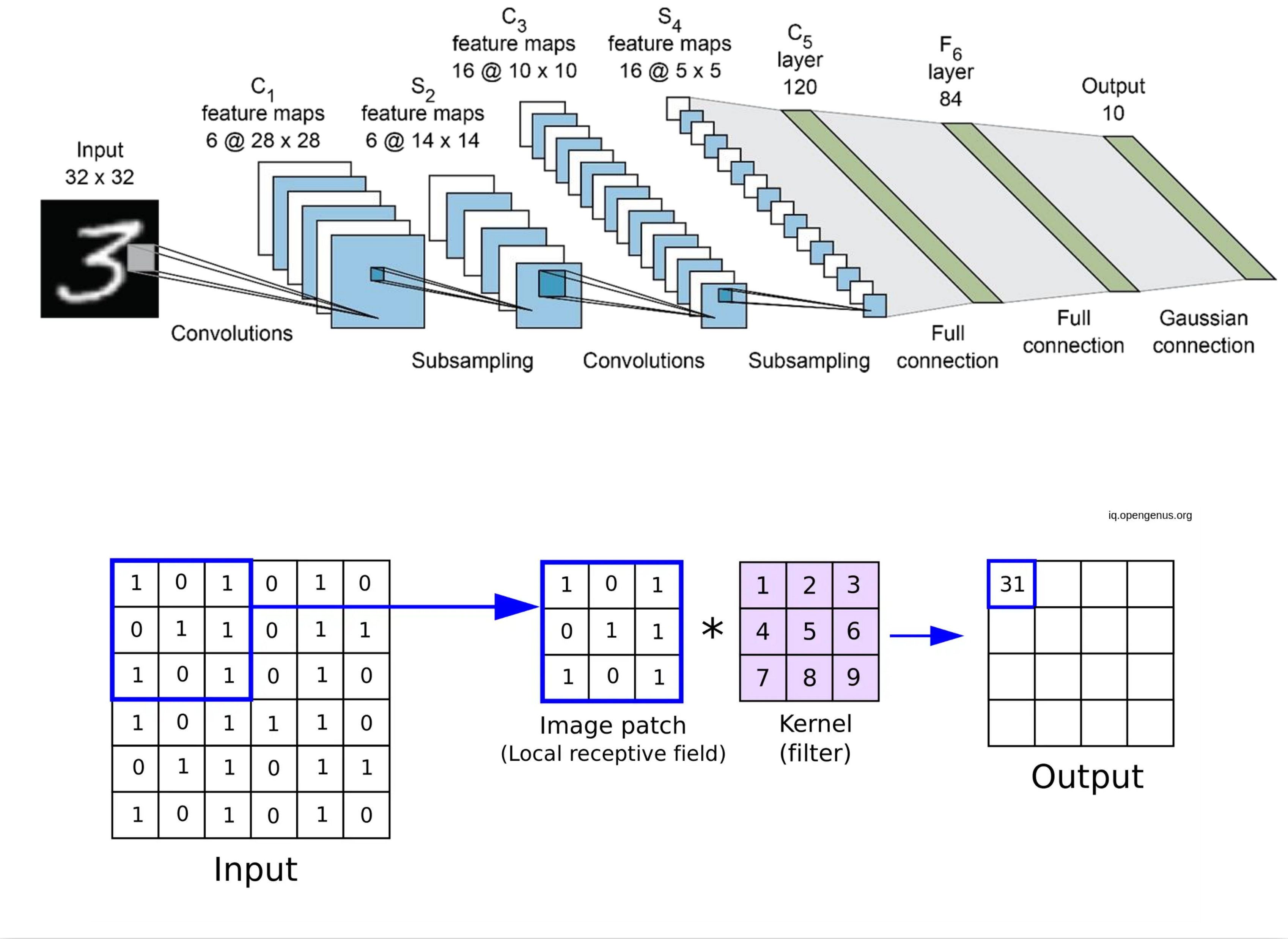
the neural network in its function becomes a linear regression.





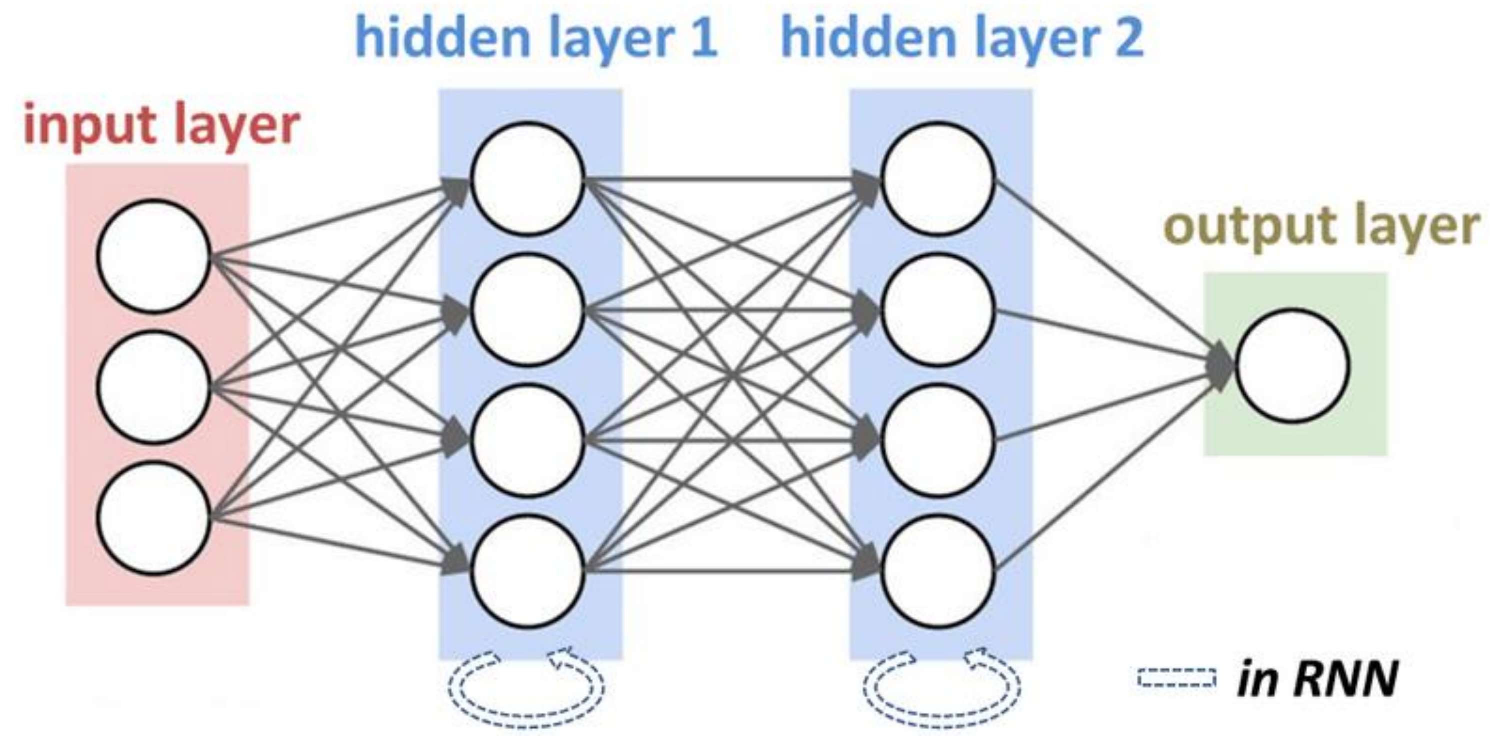
task-specific types of neural networks

[Neural Networks: Solving Complex Science Problems (neilsahota.com)]



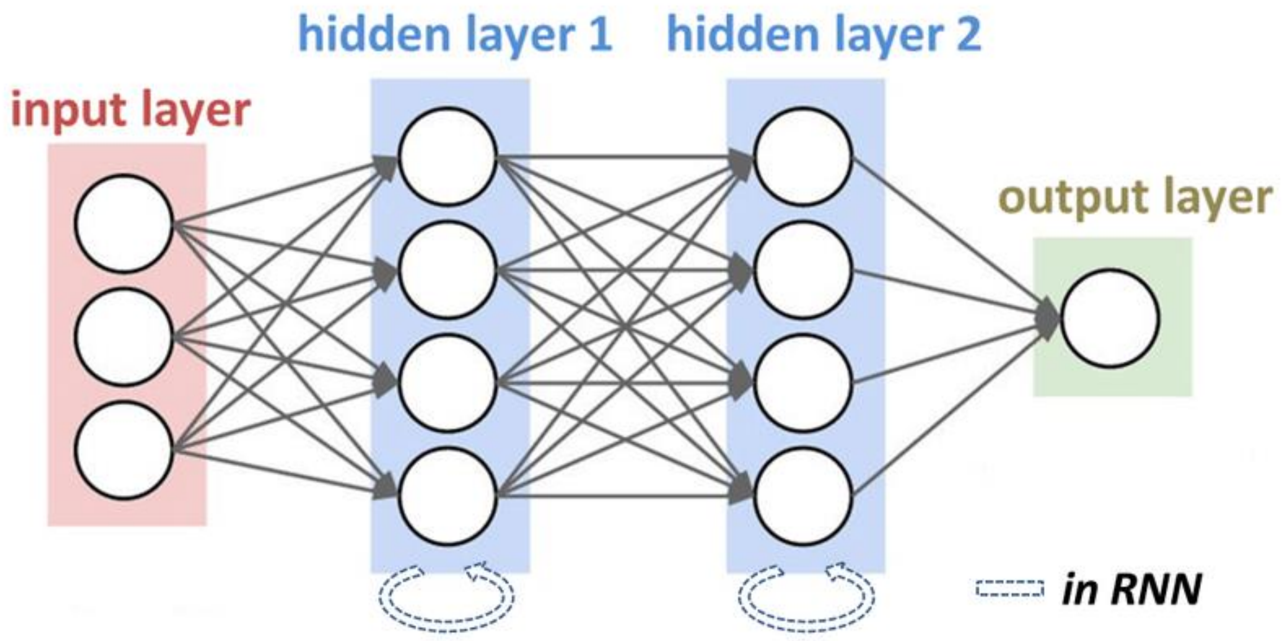
convolutional neural network

[[Convolutional Neural Networks: 1998-2023 Overview | SuperAnnotate]]



text = sequential processing

RNNs = recurrent neural networks

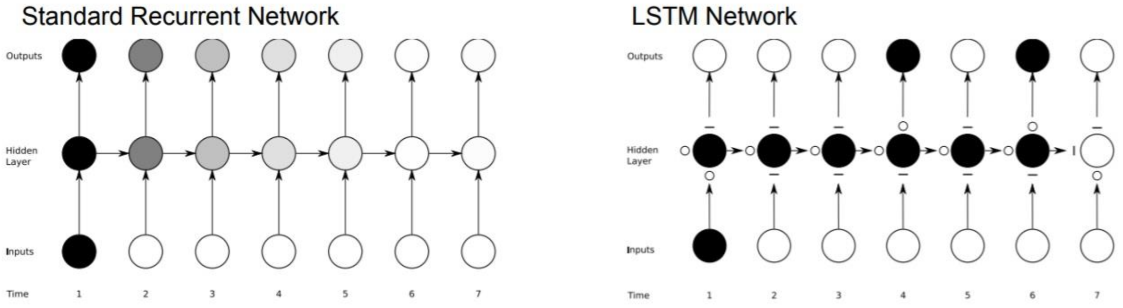


problem: vanishing gradient

RNNs

long short-term memory LSTMs

LSTMs reduce vanishing gradient problem



Graves et al 2013

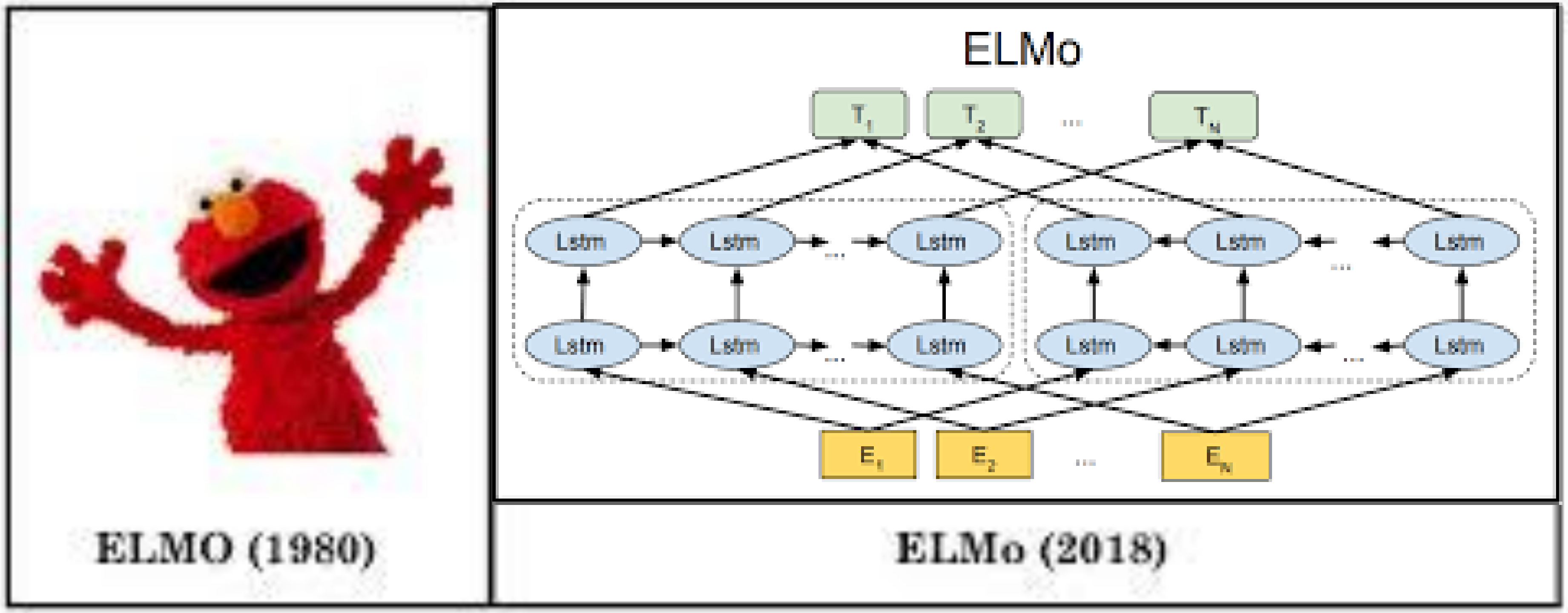
- The darker the shade, the greater the sensitivity
- The sensitivity decays exponentially over time as new inputs overwrite the activation of hidden unit and the network ‘forgets’ the first input

why transformers ?!

welcome to the world of transfer learning

ELMo

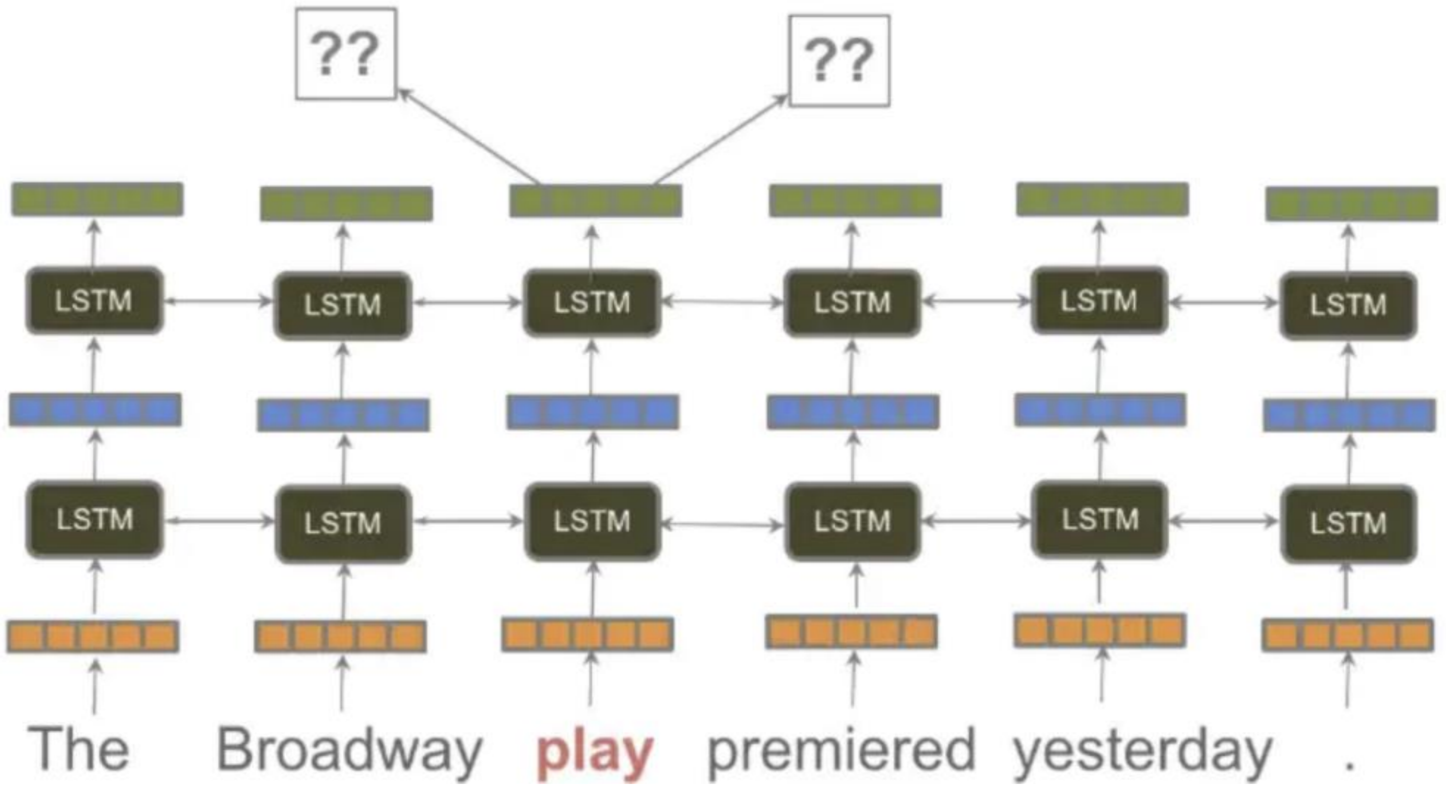
Embeddings from Language Models



ELMO (1980)

ELMo (2018)

[Peters et al. 2018]



ELMo

bidirectional

stacked

LSTM

ELMo shortcomings

- lack of ability to capture long range dependencies and global context (dynamic context window)
- training complexity
  - computationally expensive and time-consuming
- fixed architecture
  - not adaptable to different tasks and data sizes
- lack of masked language modeling

transformers

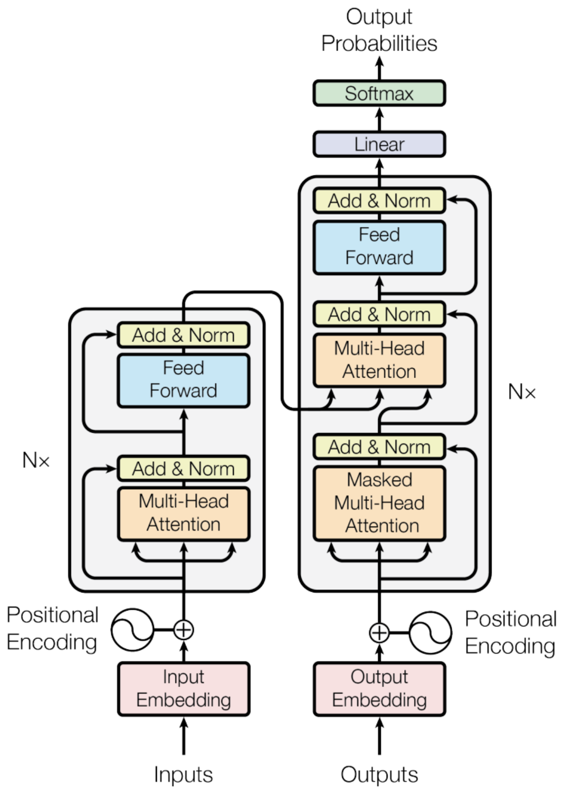
transformer architecture

- Parallel processing: Increases performance and scalability

- Bidirectionality: Allows understanding of ambiguous words and coreferences

BERT

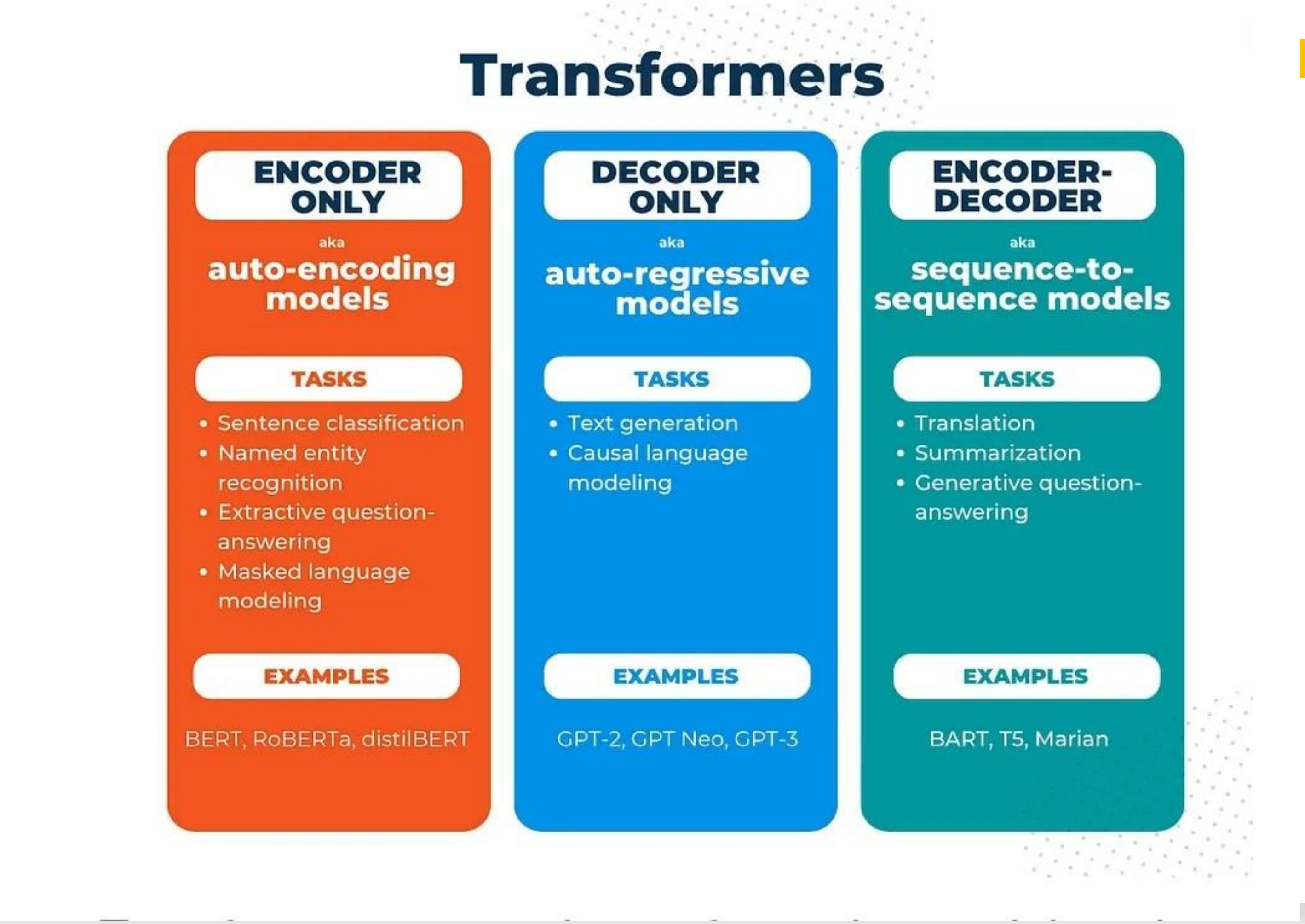
Encoder



GPT

Decoder

transformer architecture



attention

A mechanism that computes a weighted sum of values (elements in a sequence) based on their similarity to a given query.

It allows the model to dynamically emphasize or de-emphasize different elements in the sequence, enabling the capture of contextual information and relationships between elements.

[1706.03762] Attention Is All You Need (arxiv.org)

a little help from ChatGPT

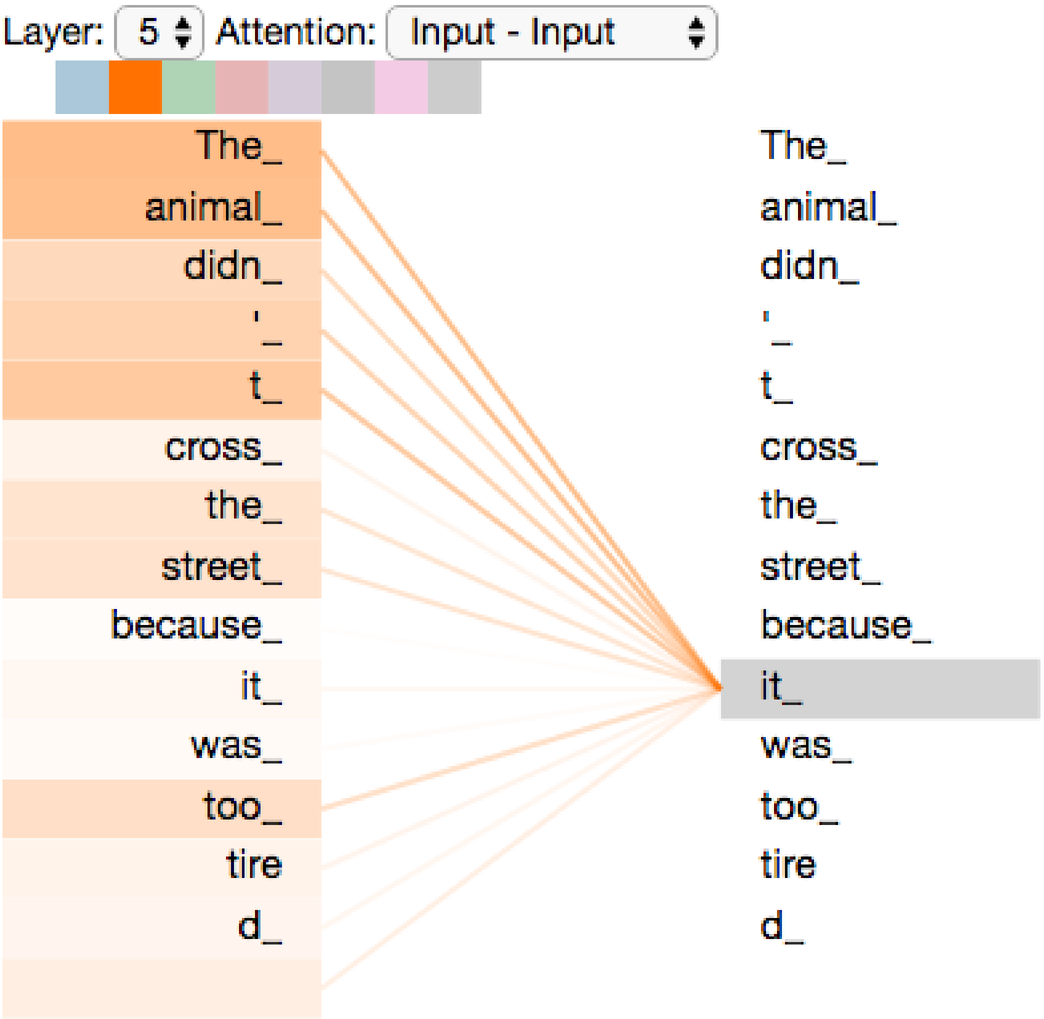
Attention Mechanism in Transformers

(in Simple Terms):

Imagine you're reading a sentence, and you want to understand the meaning of a specific word. Instead of just looking at that one word, you glance at all the words in the sentence. However, you pay more attention to the words that are more relevant to the word you're trying to understand. This way, you get a better grasp of the context and meaning.

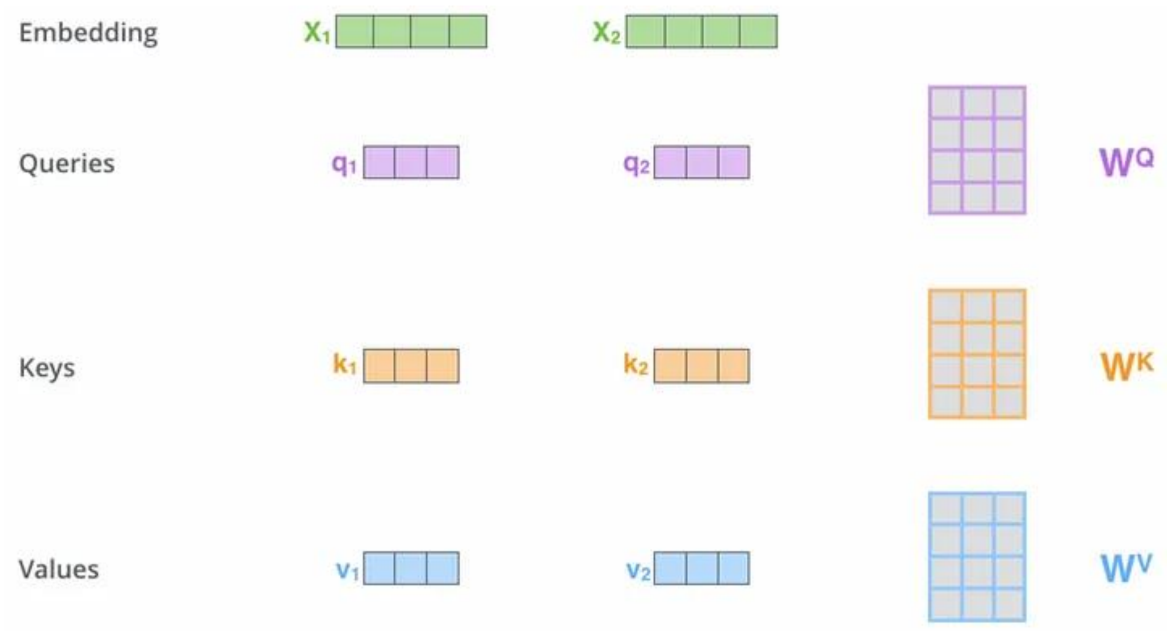
The attention mechanism in Transformers works similarly. It helps the model focus on different parts of a sentence when trying to understand a specific word. It's like giving more importance to the words that matter most for understanding that word's context. This allows the model to create richer and more accurate representations of words and their relationships.

[bertviz_tutorial.ipynb - Colaboratory (google.com)]



attention score

needs query and key



attention score needs query and key

why attention

- **capturing context**
- **handling long-range dependencies**
- **parallel processing**
- **interpretable representations**

[Attention for Neural Networks, Clearly Explained!!! - YouTube]