

Self Supervised Visual Feature Learning with Deep Neural Nets

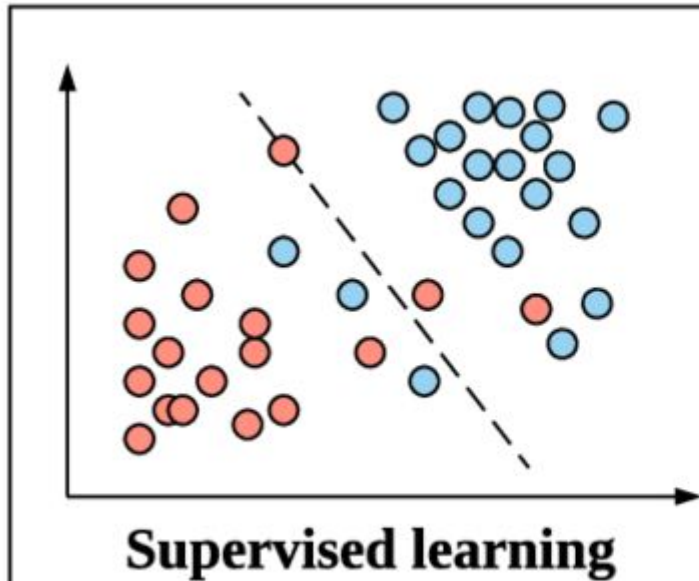
...

Rajesh Shreedhar Bhat
Data Scientist @Walmart Labs, Bengaluru

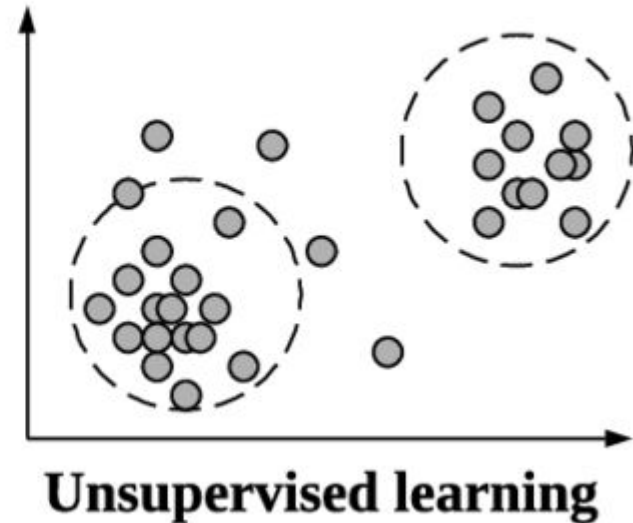
Agenda

- Different machine learning paradigms.
 - Supervised, Un-supervised, Sem-supervised, Transfer Learning, Self-Supervised Learning
- Why self-supervised learning ?
- Terminologies in Self-Supervised Learning tasks.
- Pre-text tasks.
 - RotNet.
 - Selfie(will cover BERT and later discuss about Selfie), etc ..

Different machine learning paradigms

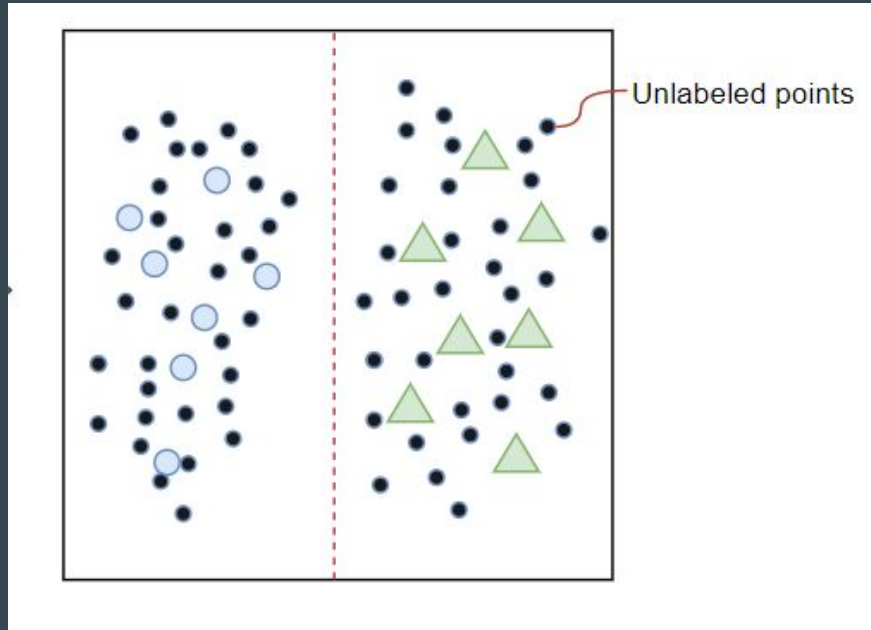


Training data + associated labels (x, y)

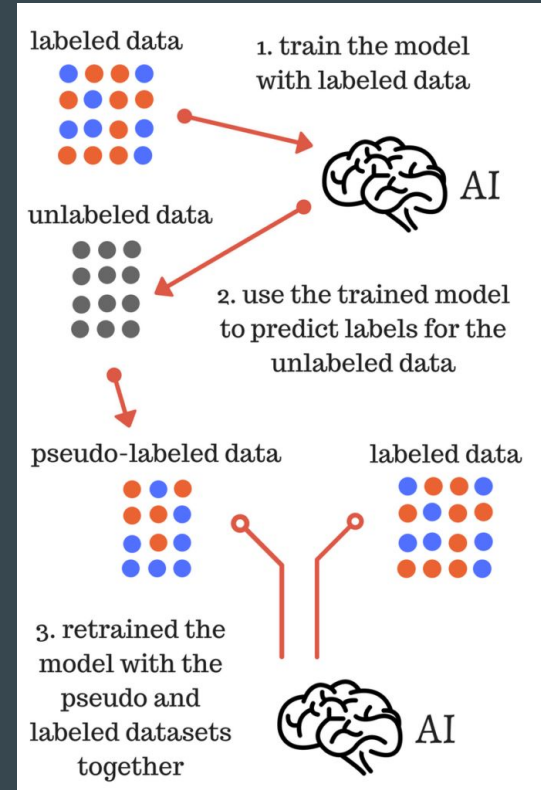


Training data (x) without labels

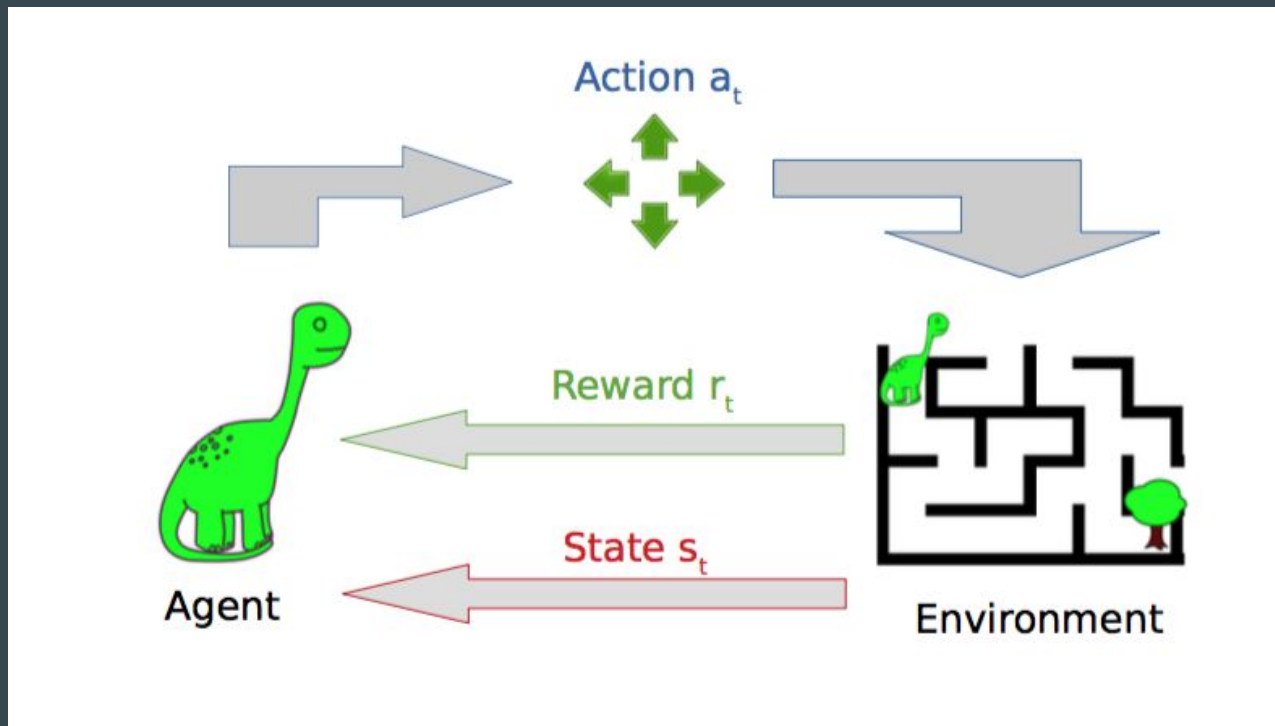
Semi-supervised learning



Small labelled set(x, y) + large unlabeled set (x)



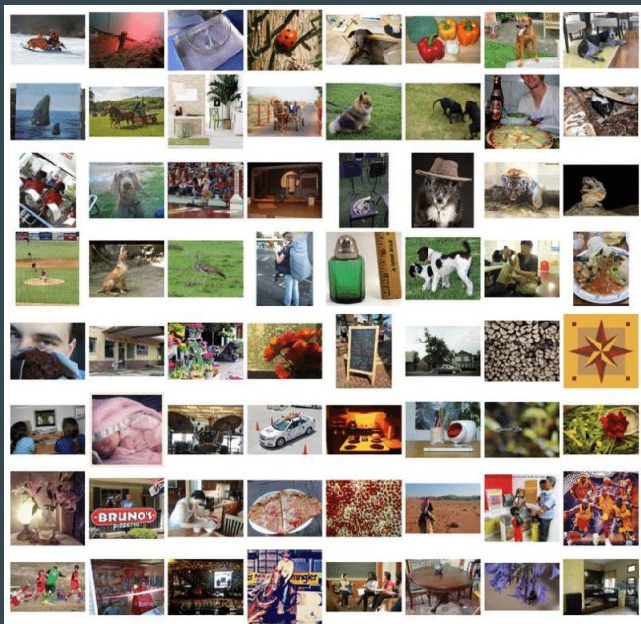
Reinforcement Learning



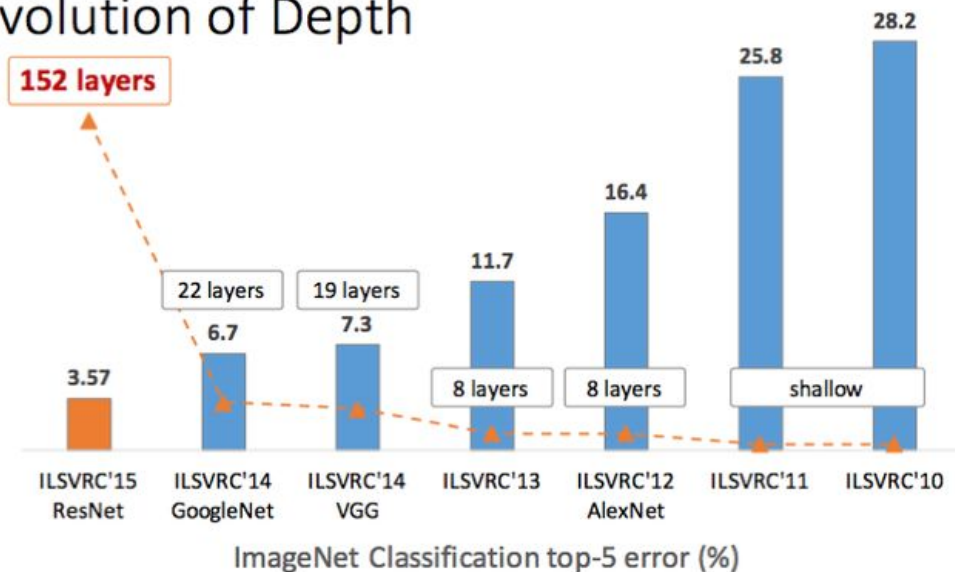
Training an agent using reward system.

ImageNet Challenge Story ..

- 1000 categories, ~1000 samples for each category.
- Strong supervision



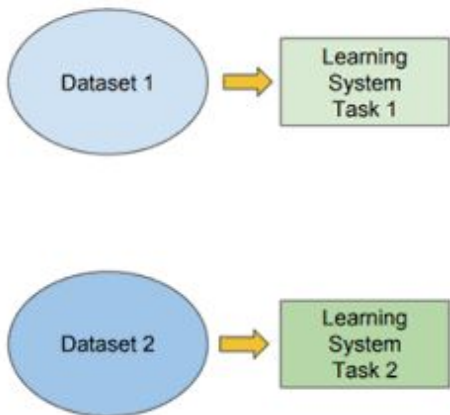
Revolution of Depth



Transfer Learning

Traditional ML

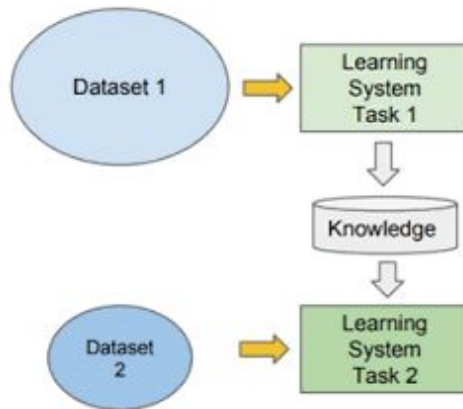
- Isolated, single task learning:
 - Knowledge is not retained or accumulated. Learning is performed w.o. considering past learned knowledge in other tasks



vs

Transfer Learning

- Learning of a new tasks relies on the previous learned tasks:
 - Learning process can be faster, more accurate and/or need less training data

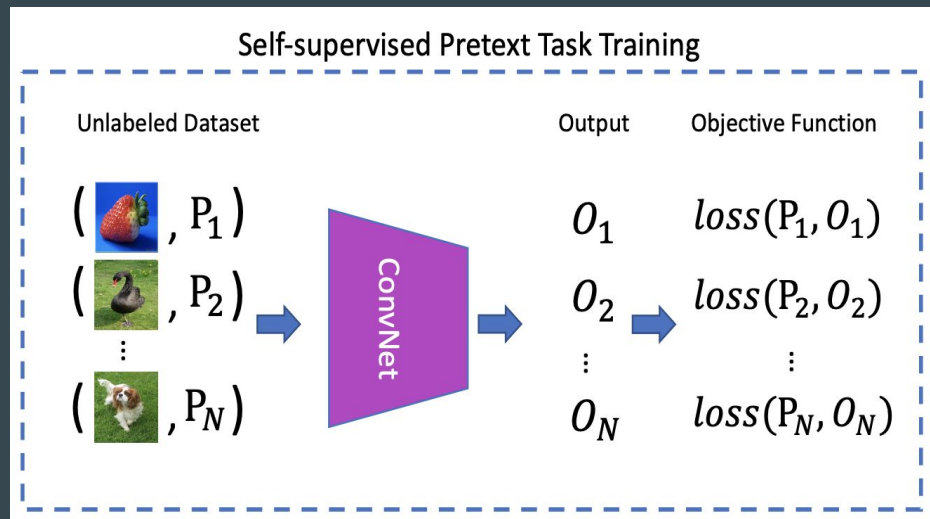


Transfer Learning ..

- Aim to start your neural network training with a pre-trained model, and fine tune it.
- Don't start with random weights, because you're starting with a model that doesn't know how to do anything at all!!
- With pre-training, you can use 1000x less data than starting from scratch.
- What if there are no pre-trained models in a particular domain ??
 - Should we start training the model from scratch ?
- Amount of improvement from an ImageNet pretrained model when applied to medical imaging is not that great.
- We need something which works better but doesn't need a huge amount of data for training. **The secret is “self-supervised learning”.**

Self-Supervised Learning

Self-supervised Learning



Train a model using labels that are naturally part of the input data, rather than requiring separate external labels(human annotated)

Why self-supervised learning ?

- Cost involved in generating new labelled dataset for every new task.
- Some domains are supervision starved e.g. medical data.
- Availability of vast number of unlabelled images/videos.
 - Facebook: millions of images uploaded per day.
 - ~300 hours of video are uploaded to YouTube every minute.

SSL: Terminologies

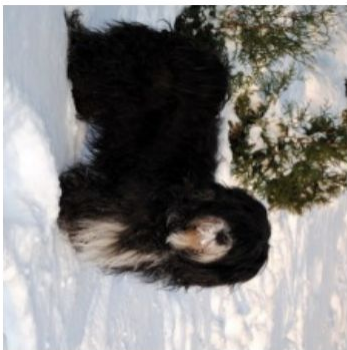
- In self-supervised learning the task that we use for pre-training is known as the “**pretext task**”.
 - Example: Auto-encoders, Compression is the pretext task.
- The tasks that we then use for fine tuning are known as the “**downstream tasks**”.
 - Classification, Object detection, Image Segmentation.

Commonly used pretext tasks in SSL in Computer Vision

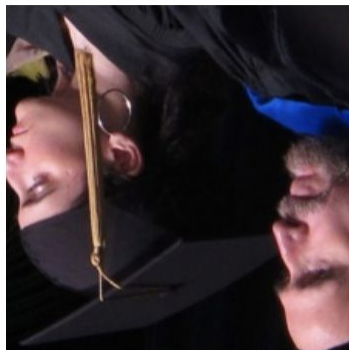
RotNet, ICLR 2018



90° rotation



270° rotation



180° rotation



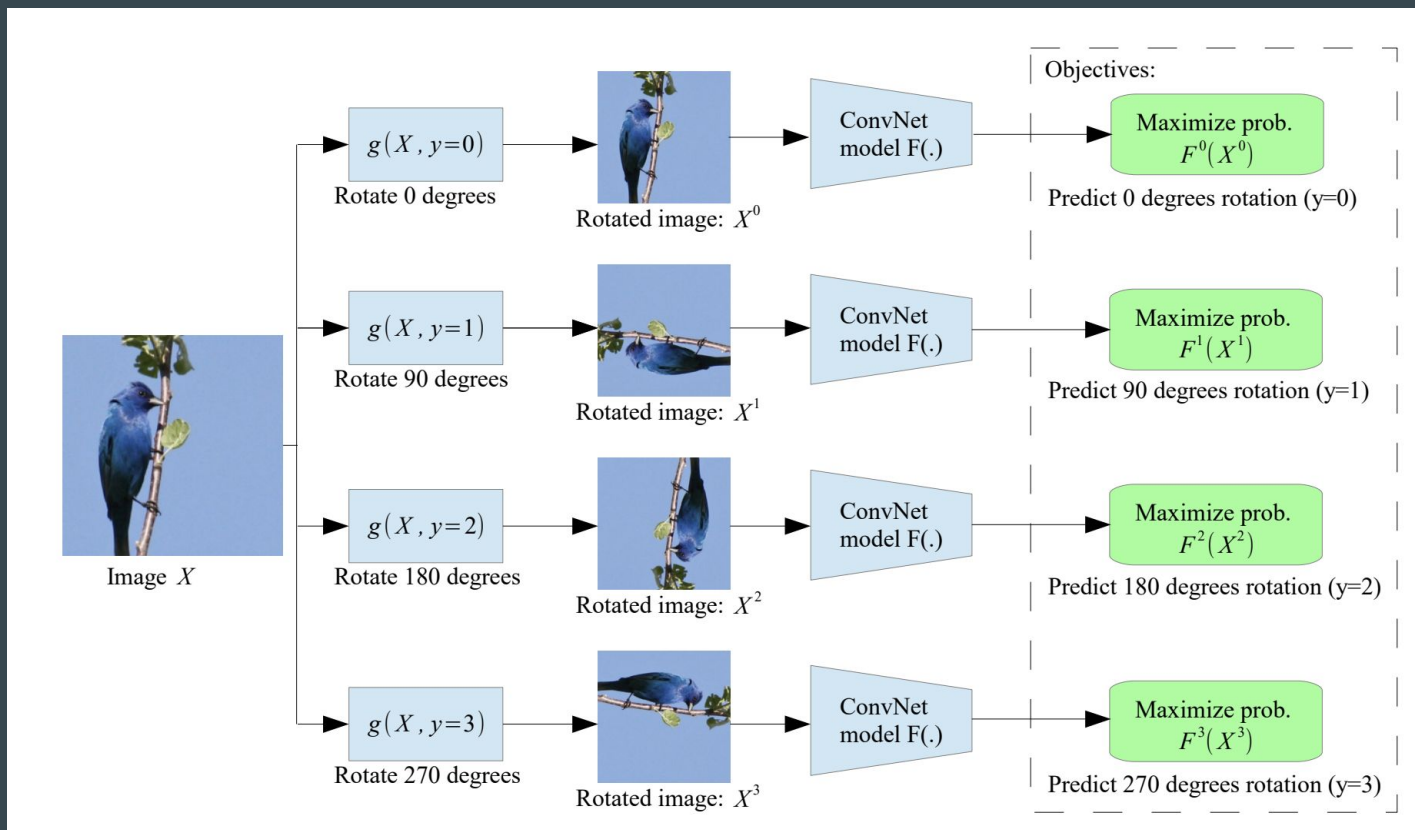
0° rotation



270° rotation

- Images rotated by random multiples of 90 degrees.
- The core intuition of RotNet authors is that if someone is not aware of the concepts of the objects depicted in the images, he/she cannot recognize the rotation that was applied to them.

RotNet ..



RotNet Results on standard datasets

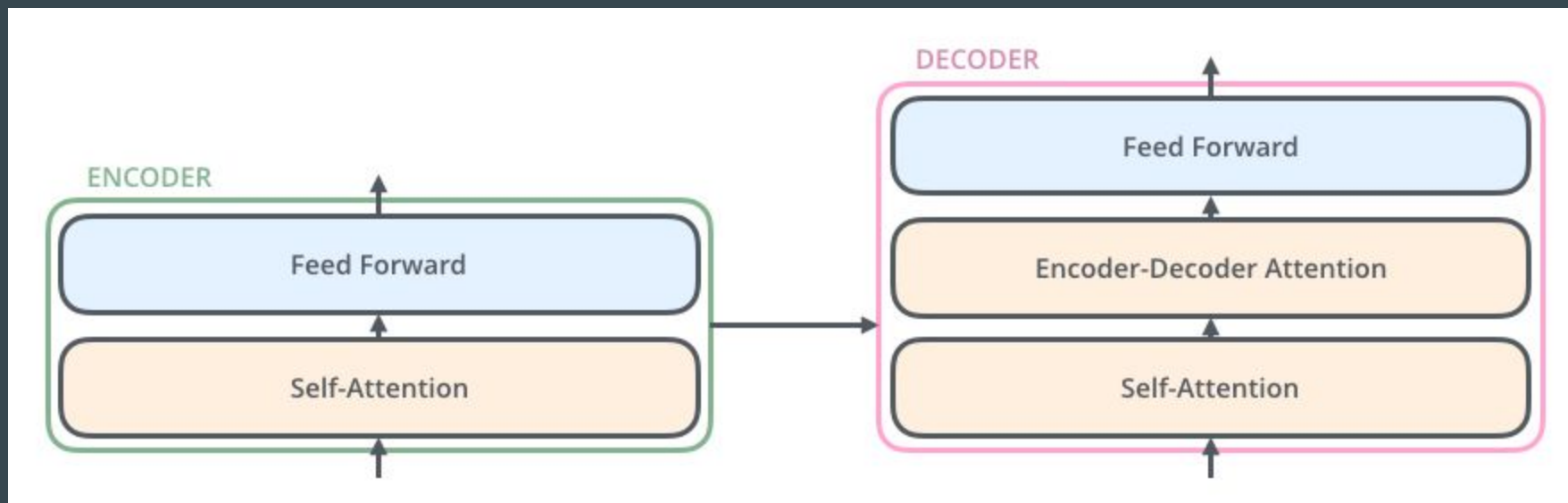
Method	Conv1	Conv2	Conv3	Conv4	Conv5
ImageNet labels	19.3	36.3	44.2	48.3	50.5
Random	11.6	17.1	16.9	16.3	14.1
Random rescaled Krähenbühl et al. (2015)	17.5	23.0	24.5	23.2	20.6
Context (Doersch et al., 2015)	16.2	23.3	30.2	31.7	29.6
Context Encoders (Pathak et al., 2016b)	14.1	20.7	21.0	19.8	15.5
Colorization (Zhang et al., 2016a)	12.5	24.5	30.4	31.5	30.3
Jigsaw Puzzles (Noroozi & Favaro, 2016)	18.2	28.8	34.0	33.9	27.1
BIGAN (Donahue et al., 2016)	17.7	24.5	31.0	29.9	28.0
Split-Brain (Zhang et al., 2016b)	17.7	29.3	35.4	35.2	32.8
Counting (Noroozi et al., 2017)	18.0	30.6	34.3	32.5	25.7
(Ours) RotNet	18.8	31.7	38.7	38.2	36.5

Selfie: **Self**-supervised pre-training for **I**mage **E**mbedding

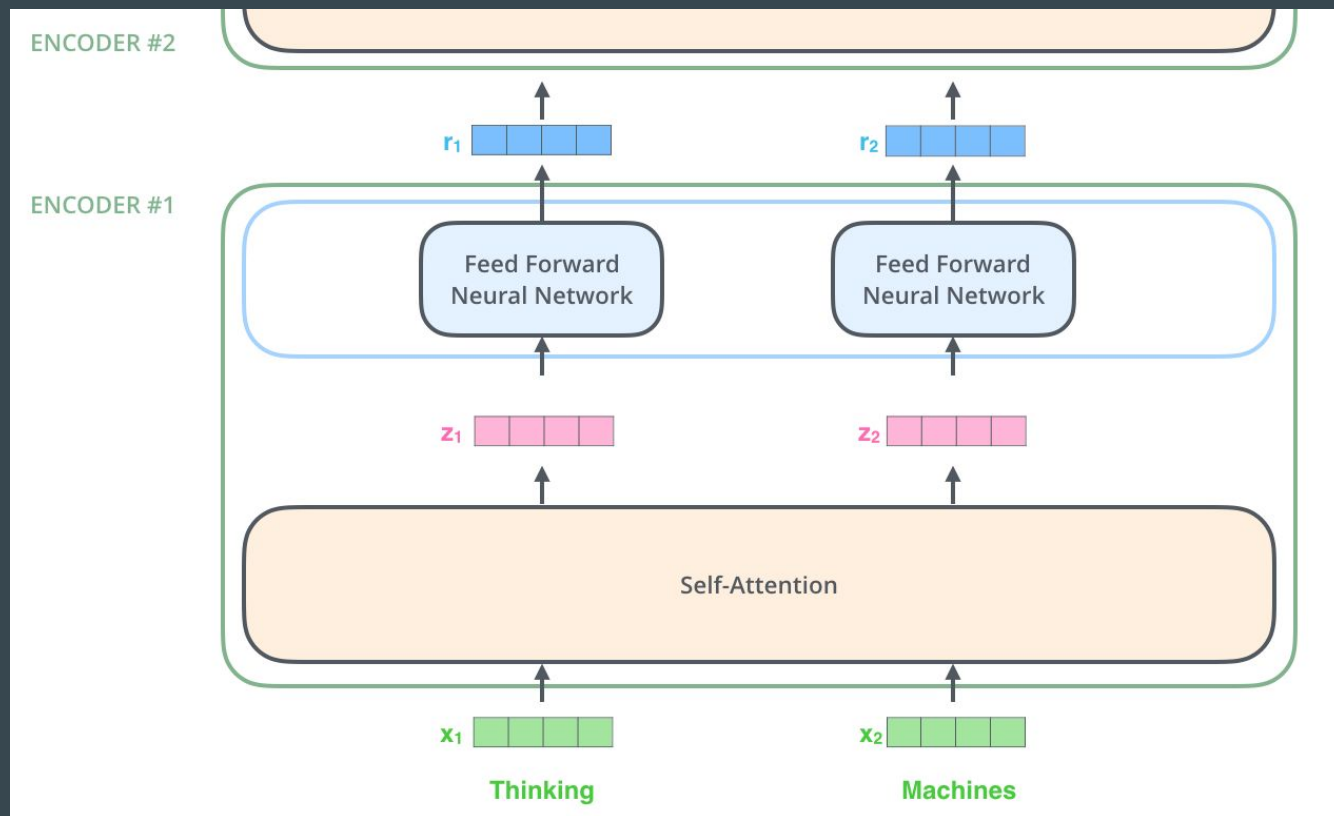
Transformers



Encoder - Decoder Network



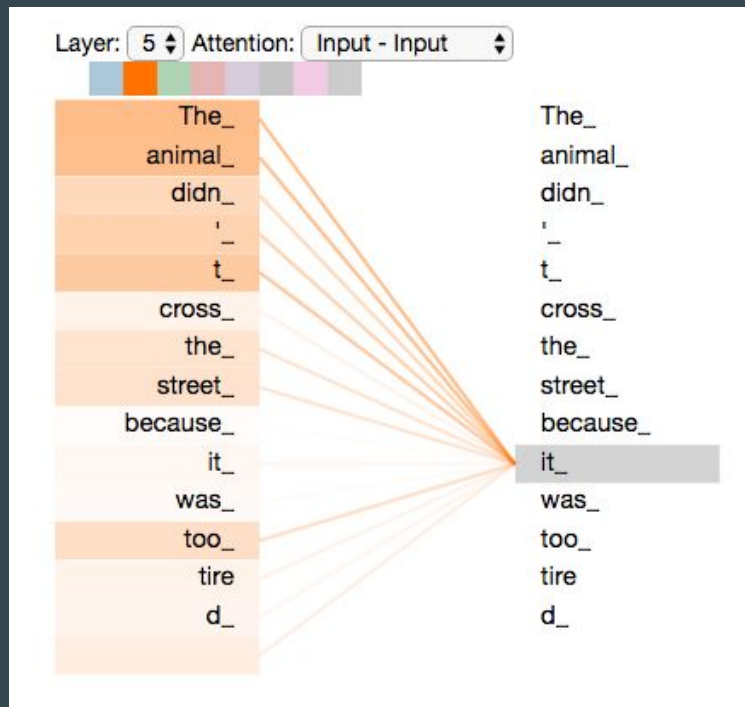
Encoder building blocks



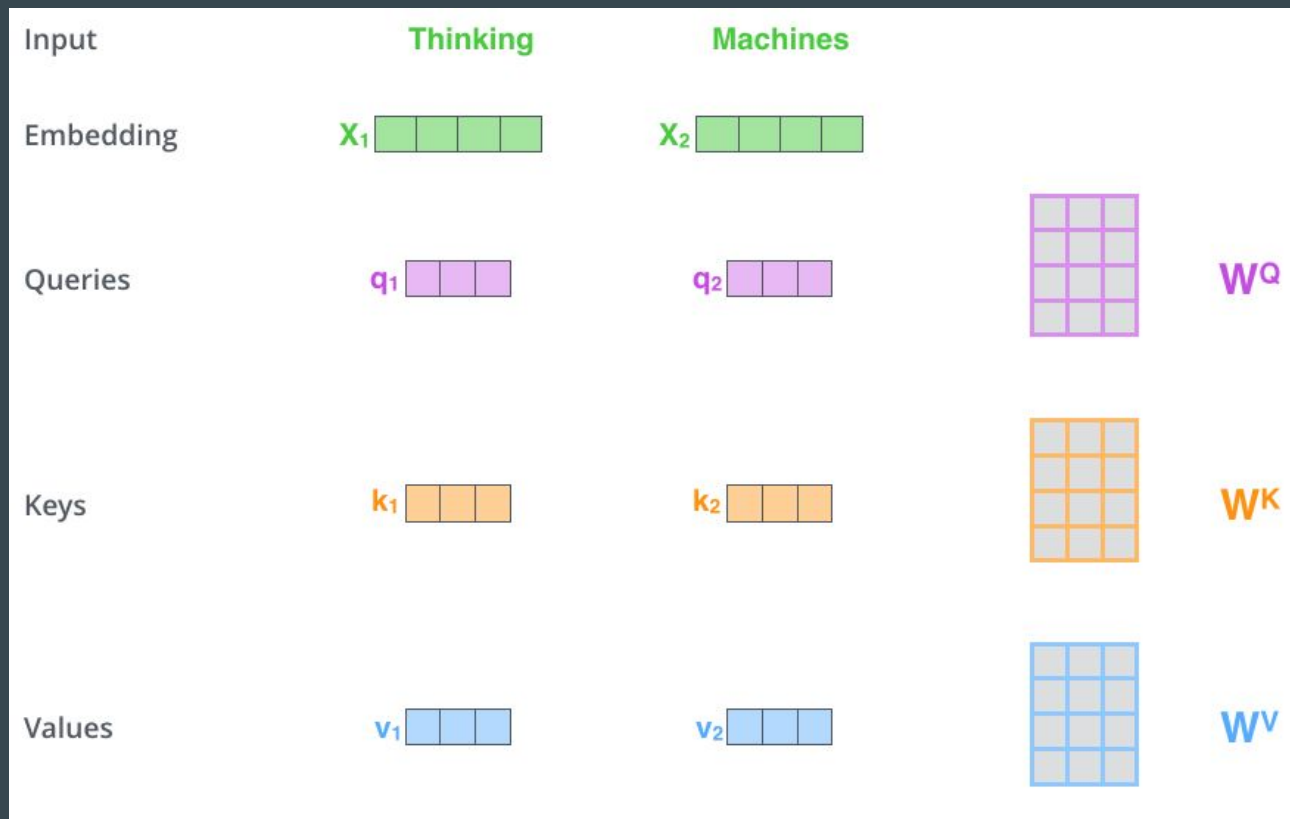
Self Attention

Translate : “The animal didn't cross the street because it was too tired”.

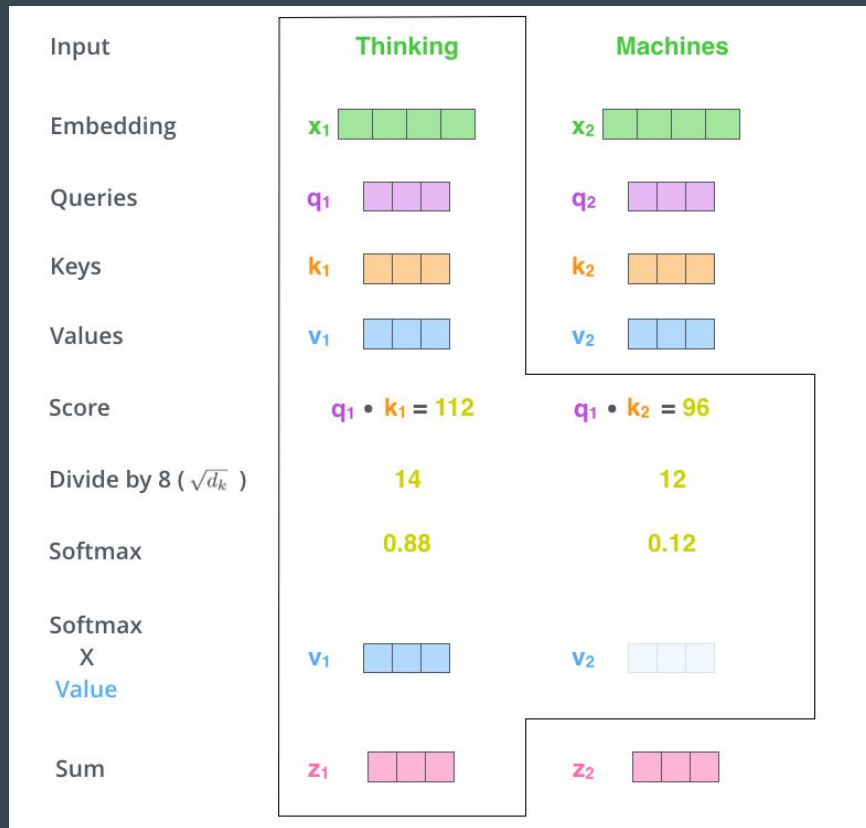
What does “it” in the above sentence refer to? Animal or Street?



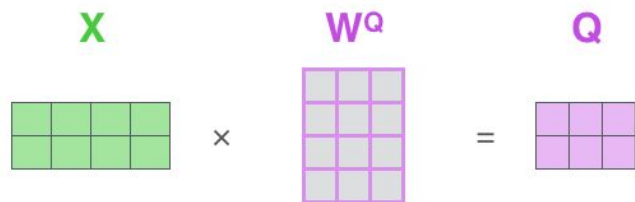
Self Attention in detail



Self-Attention: output

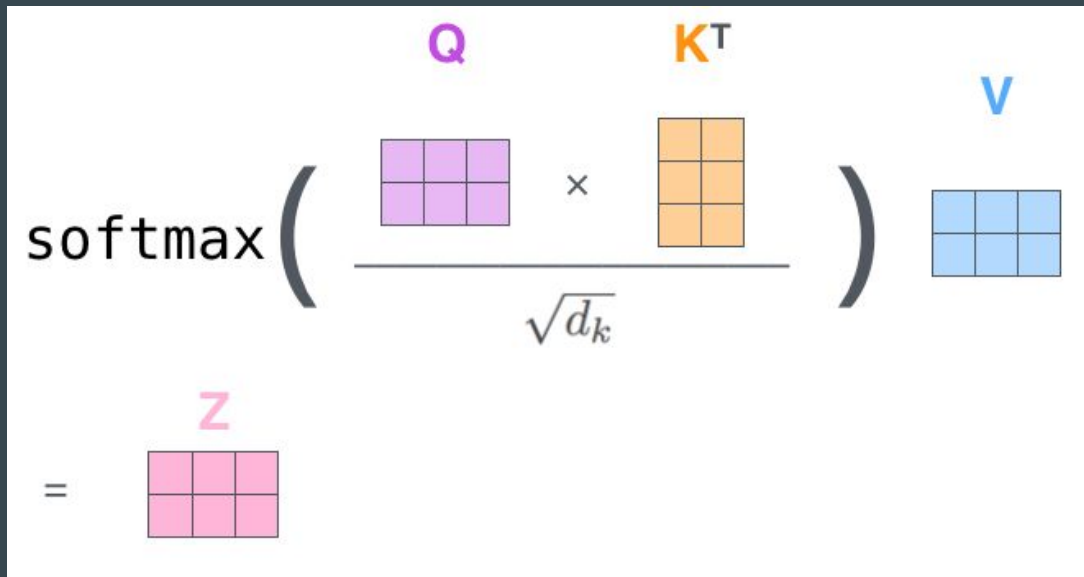


Matrix Calculation of Self-Attention

$$\mathbf{X} \times \mathbf{W}^Q = \mathbf{Q}$$


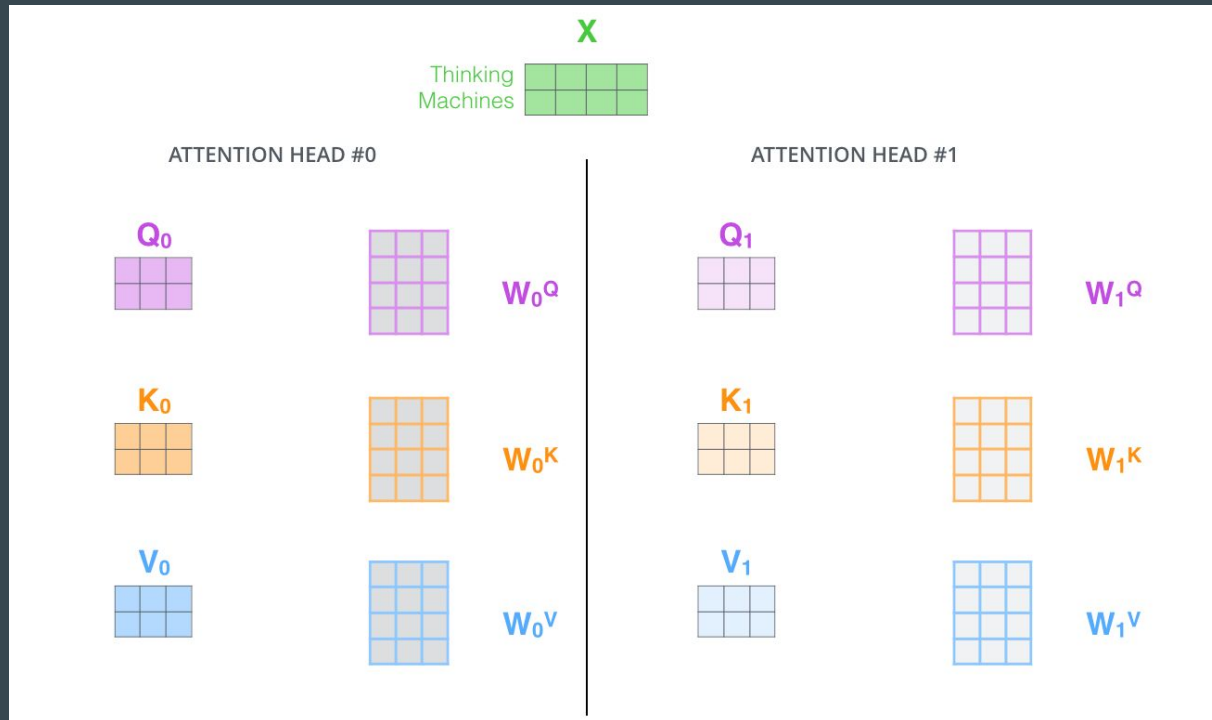
$$\mathbf{X} \times \mathbf{W}^K = \mathbf{K}$$


$$\mathbf{X} \times \mathbf{W}^V = \mathbf{V}$$


$$\text{softmax}\left(\frac{\mathbf{Q} \times \mathbf{K}^T}{\sqrt{d_k}}\right) \mathbf{V}$$

$$= \mathbf{Z}$$

Multi-head attention

- Model ability to focus on different positions.
- Input embeddings/vectors from lower encoders/decoders into a different representation subspace.



Multi-head attention ..

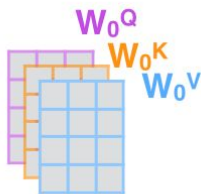
1) This is our input sentence*

Thinking
Machines

2) We embed each word*



3) Split into 8 heads.
We multiply X or R with weight matrices



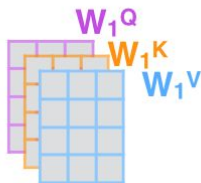
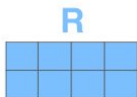
4) Calculate attention using the resulting $Q/K/V$ matrices



5) Concatenate the resulting Z matrices, then multiply with weight matrix W^O to produce the output of the layer



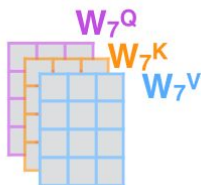
* In all encoders other than #0, we don't need embedding. We start directly with the output of the encoder right below this one



...

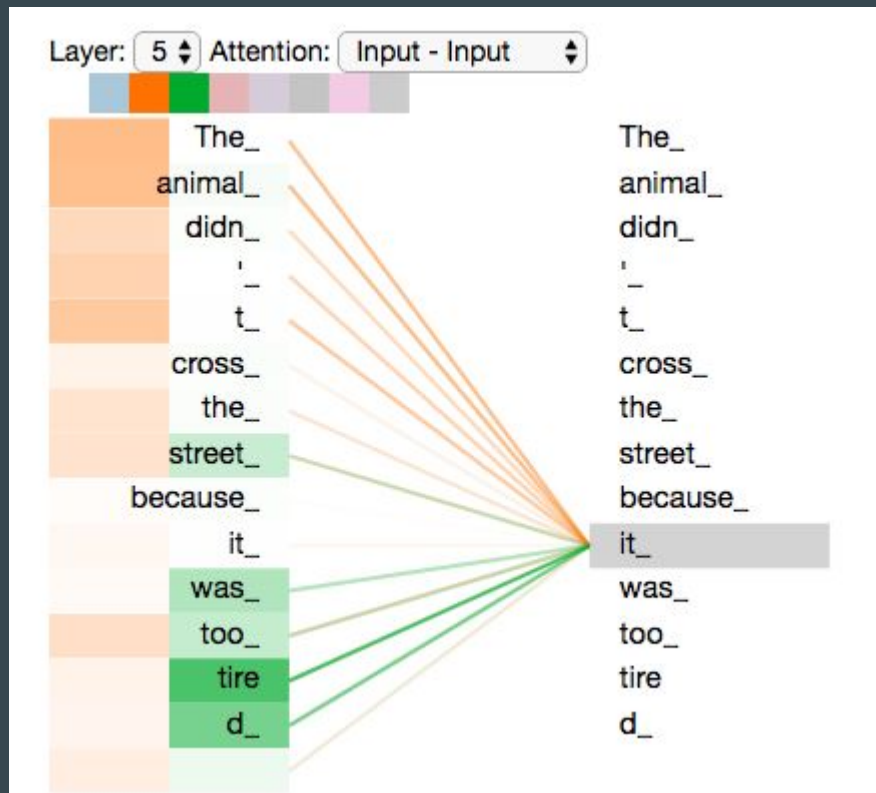
...

...



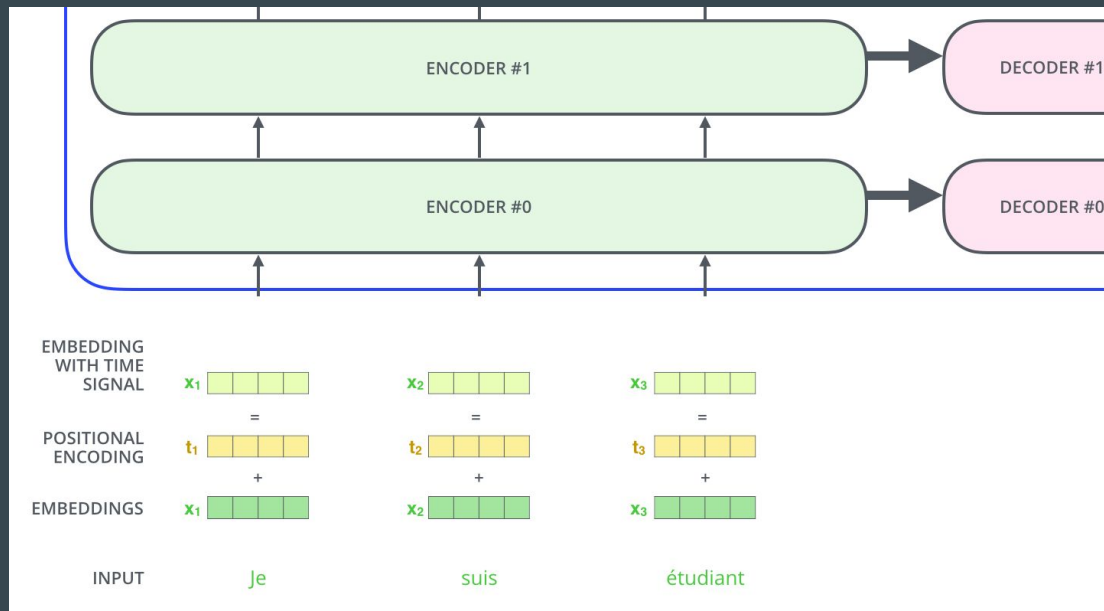
Interpreting - Multi Head Attention

- As we encode the word "it", one attention head is focusing most on "the animal", while another is focusing on "tired".
- In a sense, the model's representation of the word "it" bakes in some of the representation of both "animal" and "tired".



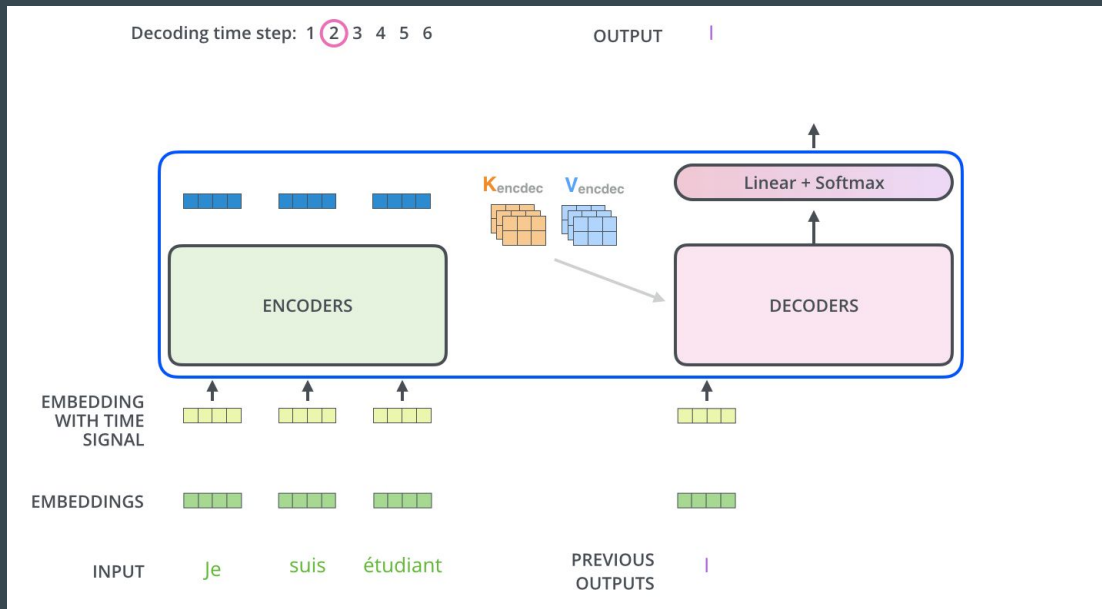
Positional Embeddings

- So far, order of the words in the input sequence is missing.
- **Positional embeddings:** vectors which follow a specific pattern that the model learns, which helps it determine the position of each word, or the distance between different words in the sequence.

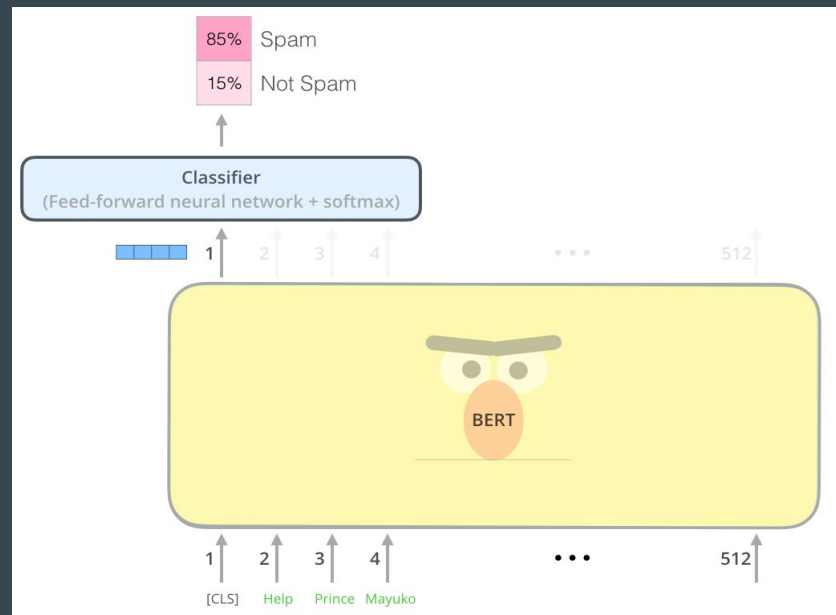
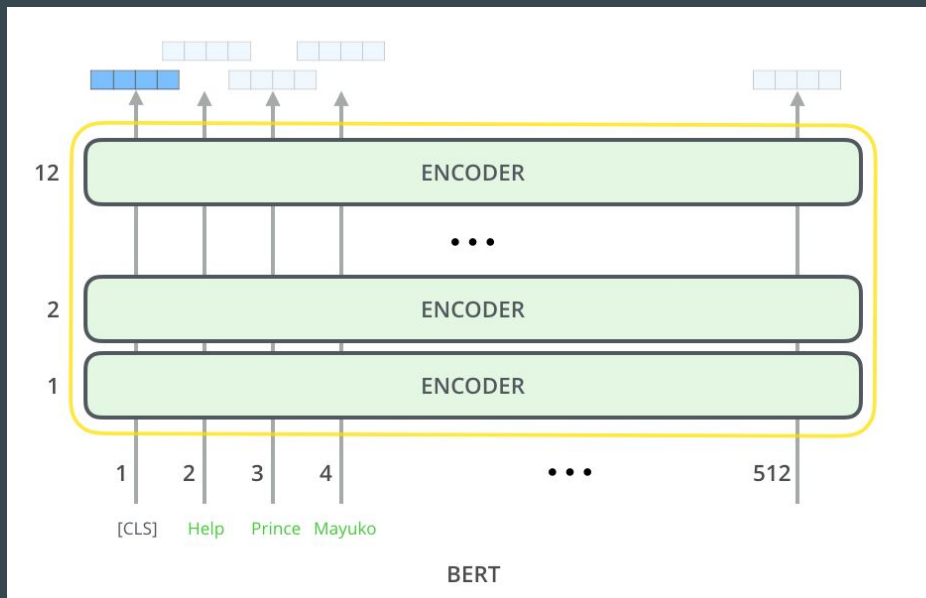


Transformer Architecture

The decoder attends on the encoder's output and its own input (self-attention) to predict the next word.



BERT: Bidirectional Encoder Representations from Transformers



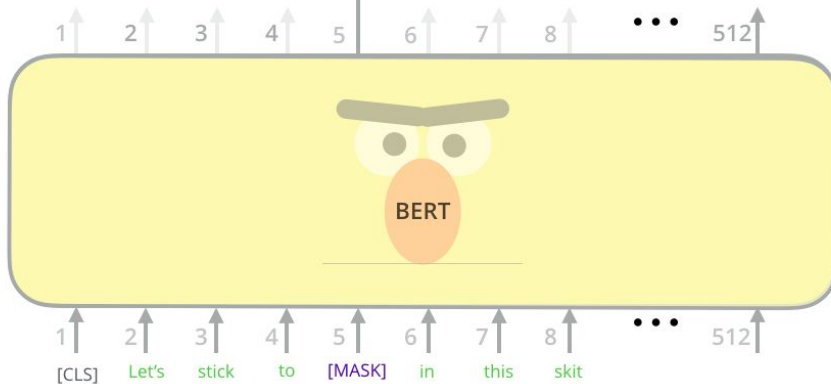
BERT: Masked language model

Use the output of the masked word's position to predict the masked word

Possible classes:
All English words

0.1%	Aardvark
...	...
10%	Improvisation
...	...
0%	Zyzyva

FFNN + Softmax



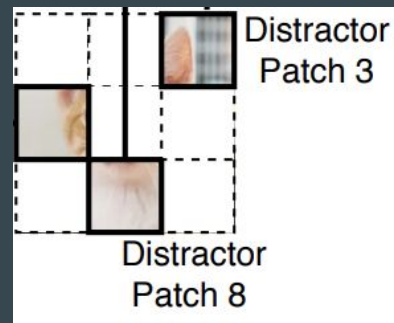
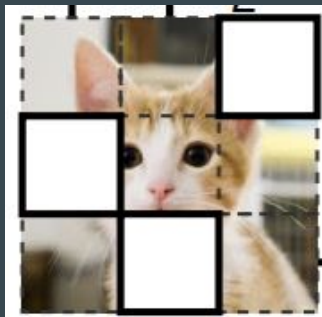
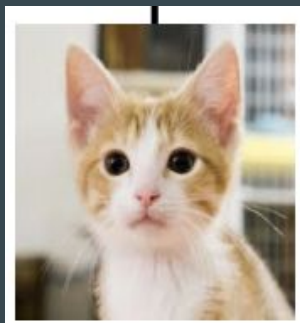
Randomly mask
15% of tokens

Input

[CLS] Let's stick to improvisation in this skit

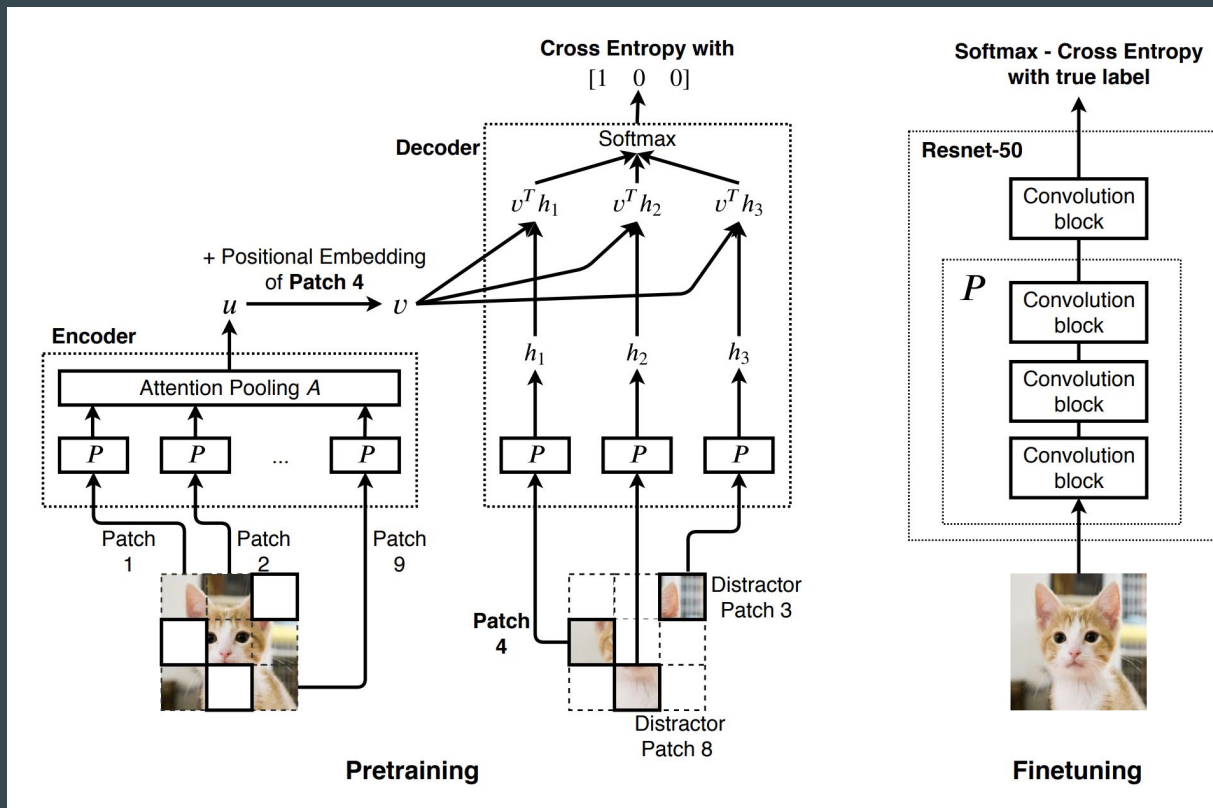
Selfie, GoogleBrain 2019

- Selfie generalizes the concept of masked language modeling of BERT (Devlin et al., 2019) to continuous data, such as images.
- Given masked-out patches in an input image, Selfie method learns to select the correct patch, among other “distractor” patches sampled from the same image, to fill in the masked location.



- To perform this task successfully, the network needs to understand the global content of the full image, as well as the local content of each individual patch and their relative relationship.

Selfie: Model Architecture



$u, \text{houtput}(1), \text{houtput}(2), \dots, \text{houtput}(n) = \text{TransformerLayers}(u_0, h_1, h_2, \dots, h_n)$

Other common pretext tasks in CV:
https://www.fast.ai/2020/01/13/self_supervised/

Choosing a pretext task ??

PPT: <https://github.com/rajesh-bhat/self-supervised-visual-representation-learning>

Questions ?



rsbhat@asu.edu



<https://www.linkedin.com/in/rajeshshreedhar>