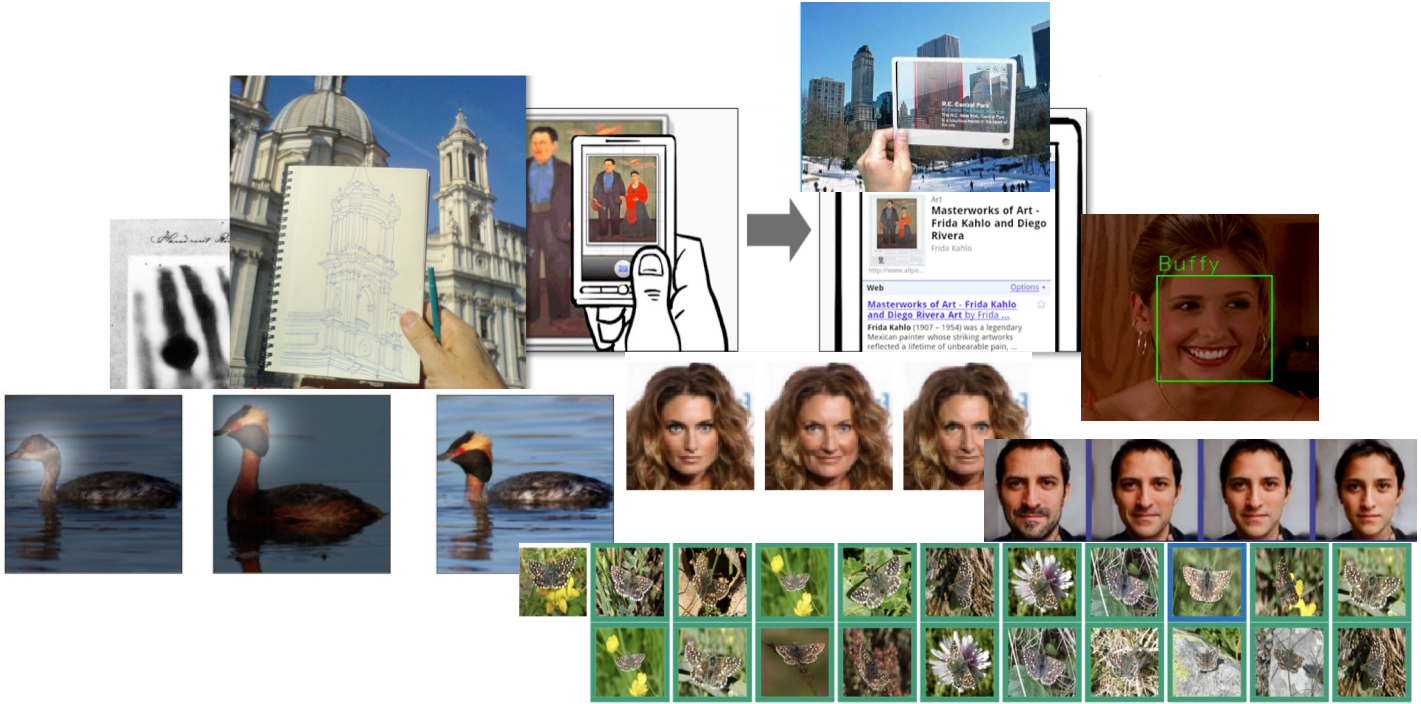


COURS RDFIA deep Image

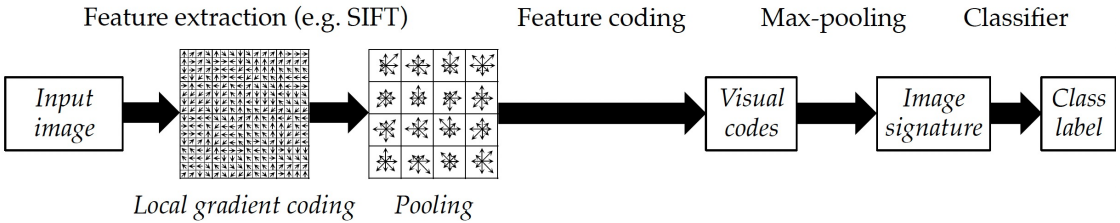
Matthieu Cord
Sorbonne University

Course Outline

1. Computer Vision and ML basics: Visual (local) feature detection and description, Bag of Word Image representation, Linear classification (SVM)
2. Introduction to Neural Networks (NNs)
3. Machine Learning basics (2): Risk, Classification, Datasets, benchmarks and evaluation
4. Neural Nets for Image Classification
5. **Vision Transformers**

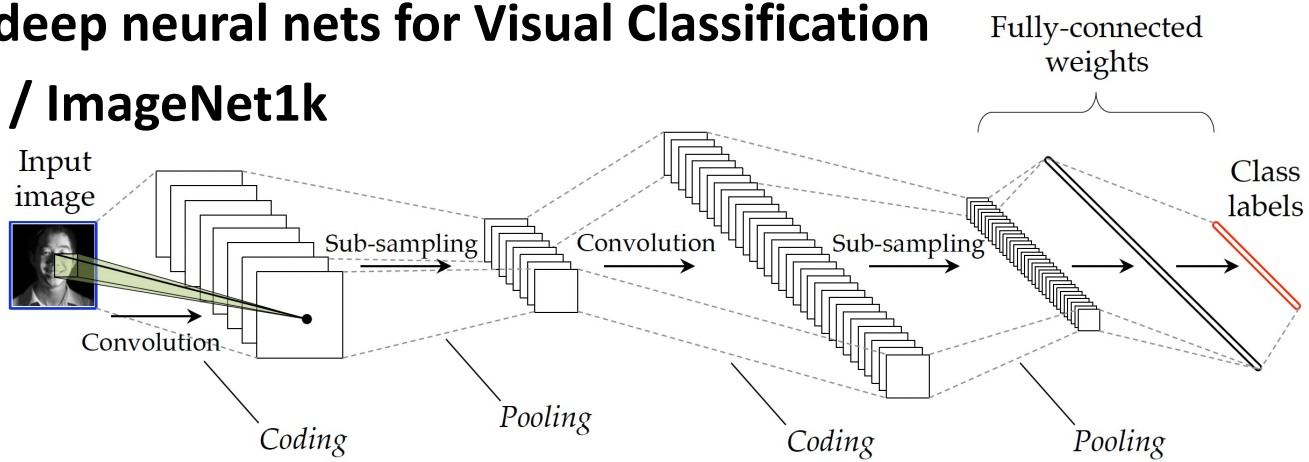
Context: Image classification **Before/After** ImageNet (2009)

The 2000s: *BoWs image modeling + SVMs* for Visual Classification



The 2010s: *Large* deep neural nets for Visual Classification

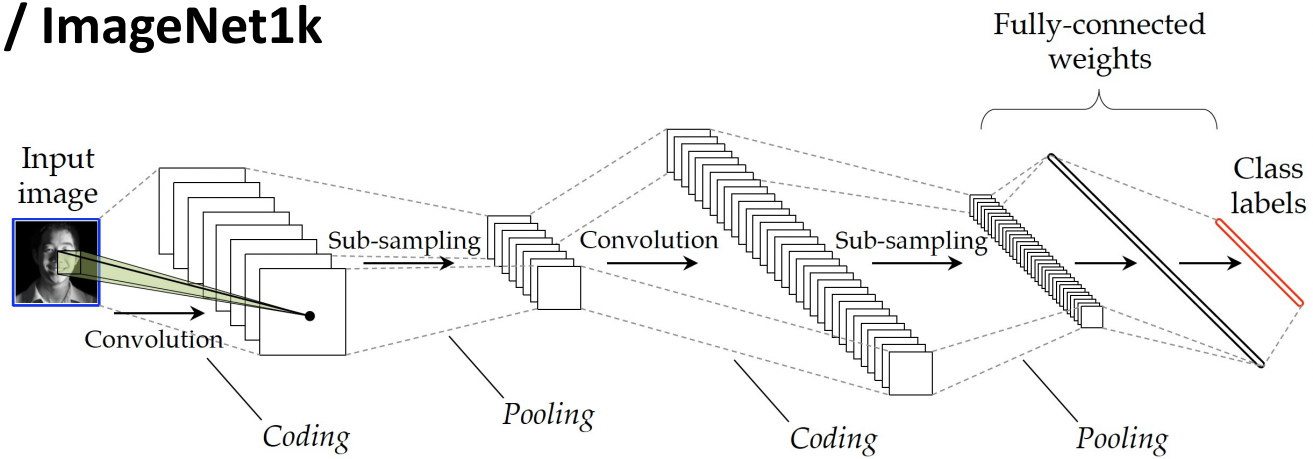
The star: **ConvNet / ImageNet1k**



Context: Image classification **After** ImageNet (2009)

The 2010s: *Large* deep neural nets for Visual Classification

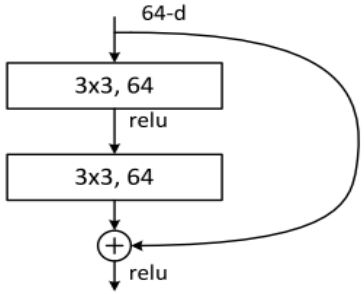
The star: **ConvNet / ImageNet1k**



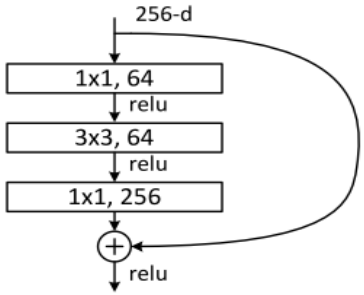
AlexNet 2012

- Same model as LeCun'98 but:
  - Bigger model (8 layers)
  - More data (10^6 vs 10^3 images)
  - GPU implementation (50x speedup over CPU)
  - Better regularization (DropOut)

Post-2012 revolution: ResNet Architecture

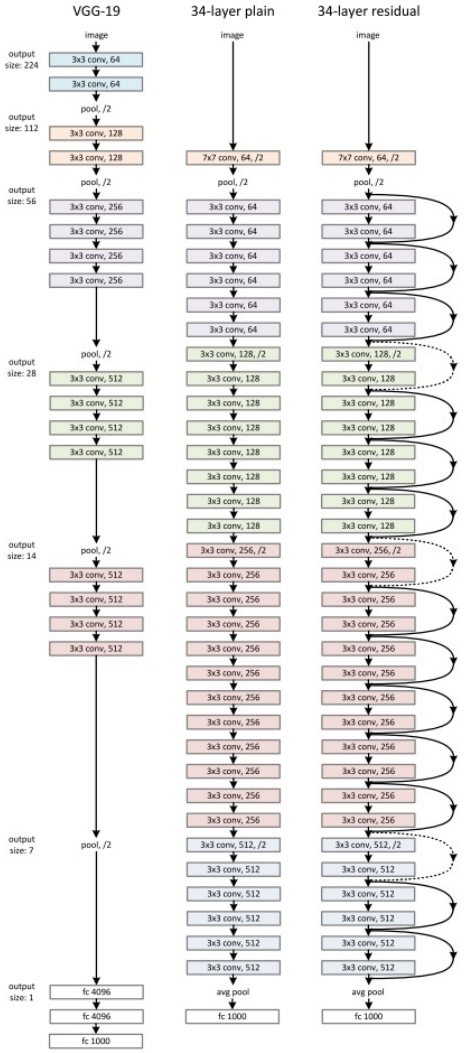


A naïve residual block



"bottleneck" residual block
(for ResNet-50/101/152)

ConvNet Configuration			
B	C	D	E
13 weight layers	16 weight layers	16 weight layers	19 weight layers
ut (224 × 224 RGB image)			
conv3-64 **conv3-64**	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool			
conv3-128 **conv3-128**	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool			
conv3-256 conv3-256	conv3-256 conv3-256 **conv1-256**	conv3-256 conv3-256 **conv3-256**	conv3-256 conv3-256 conv3-256 **conv3-256**
maxpool			
conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool			
conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool			
FC-4096			
FC-4096			
FC-1000			
soft-max			



Context: Beyond ImageNet?

The 2000s: *BoWs image modeling + SVMs* for Visual Classification

The 2010s: *Large* deep neural nets for Visual Classification

What is expected for the 2020s?

"Attention is all you need": **Transformers** for Vision !?

And **datasets? Internet…**

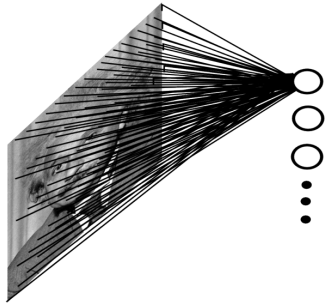
[Vaswani et al., Attention is all you need, NeurIPS 2017]

Outline

1. Computer Vision and ML basics: Visual (local) feature detection and description, Bag of Word Image representation, Linear classification (SVM)
2. Introduction to Neural Networks (NNs)
3. Machine Learning basics (2): Risk, Classification, Datasets, benchmarks and evaluation
4. Neural Nets for Image Classification
5. Vision Transformers

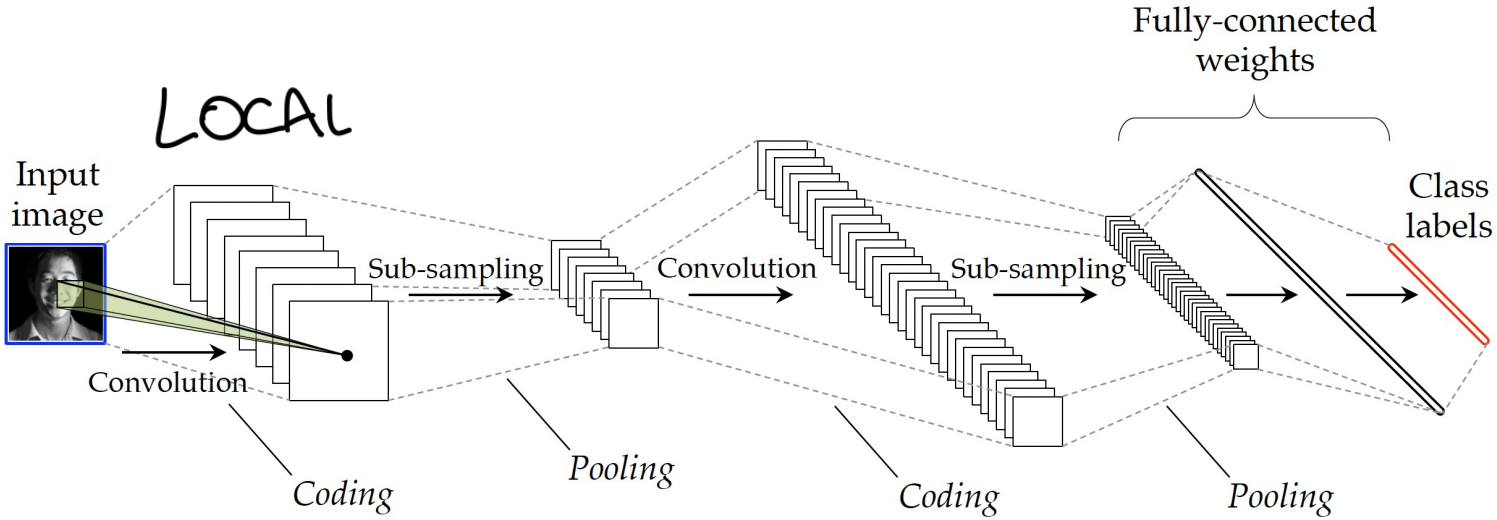
   NLP: Attention is all you need

Attention process in ConvNets



In ConvNets, what information is shared between pixels (or features) in one block? => *2D spatial locality (typically 3x3) => attention is done locally*

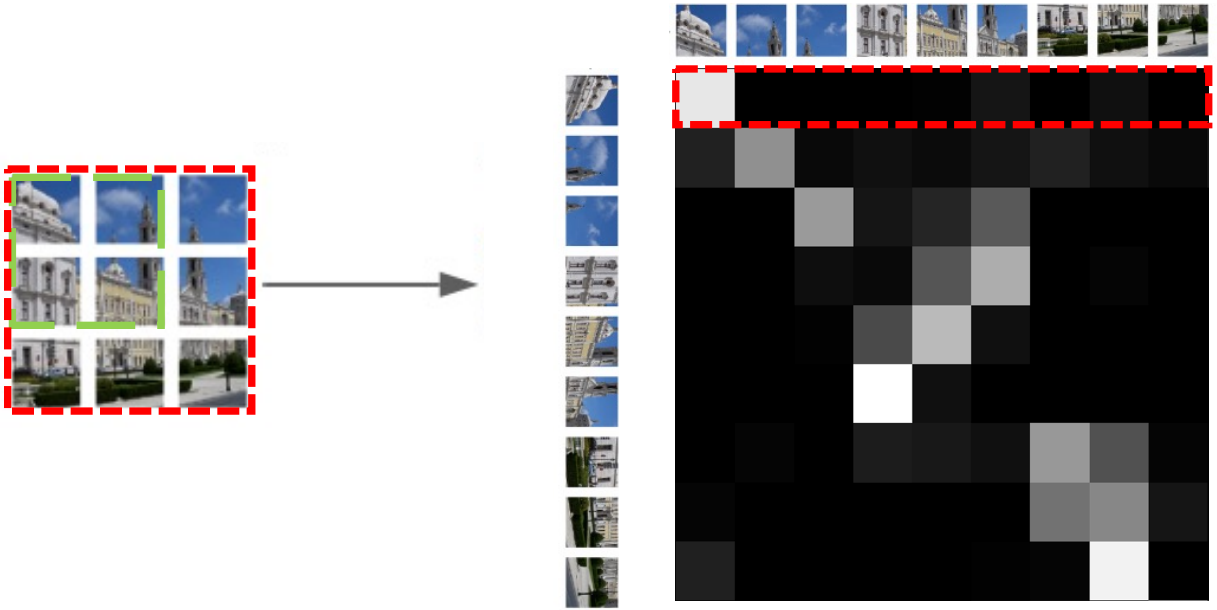
Rq: less local after many layers



Global (Self) attention

How to build a deep architecture with ~~local~~ global attention inside?
Meaning that one patch may interact with all others!

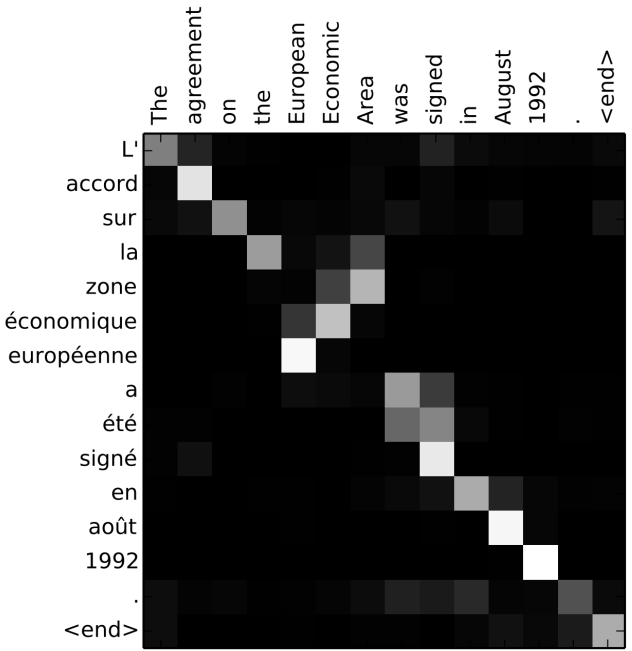
=> Different than convNet!



Let’s see what they do in Natural Language Processing (NLP):

Attention between words in Machine translation process:

1. Computing of weights
2. Use them to compute new features

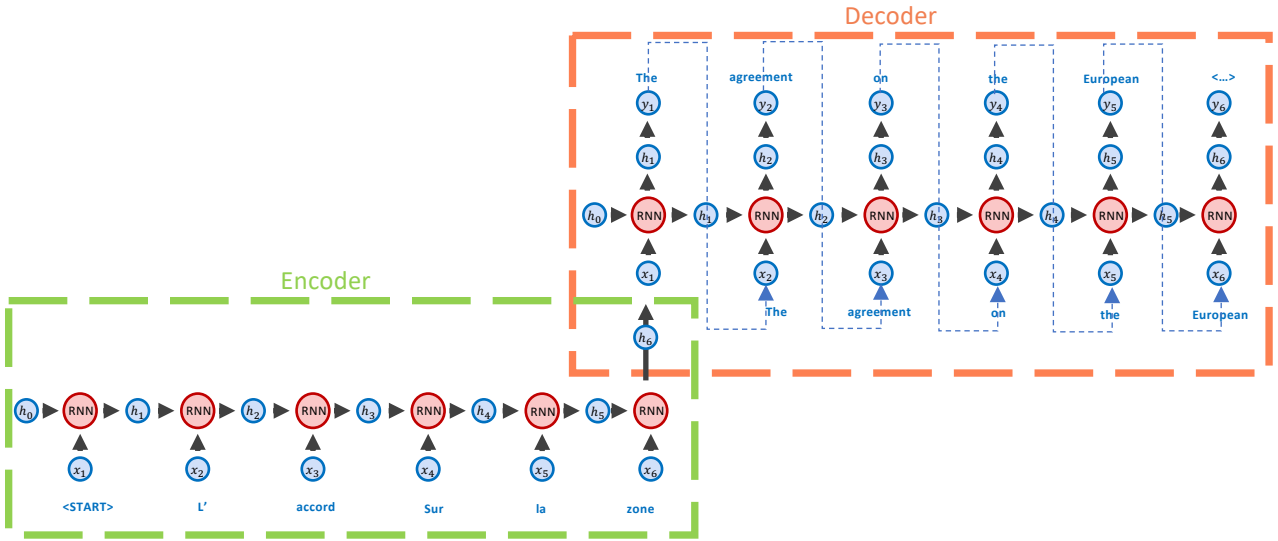


Attention process in NLP

Basic language translation models: Encoder/Decoder

Ex.: Seq2Seq -- RNNs2RNNs

Cross-attention for language translation in at the end of Encoder

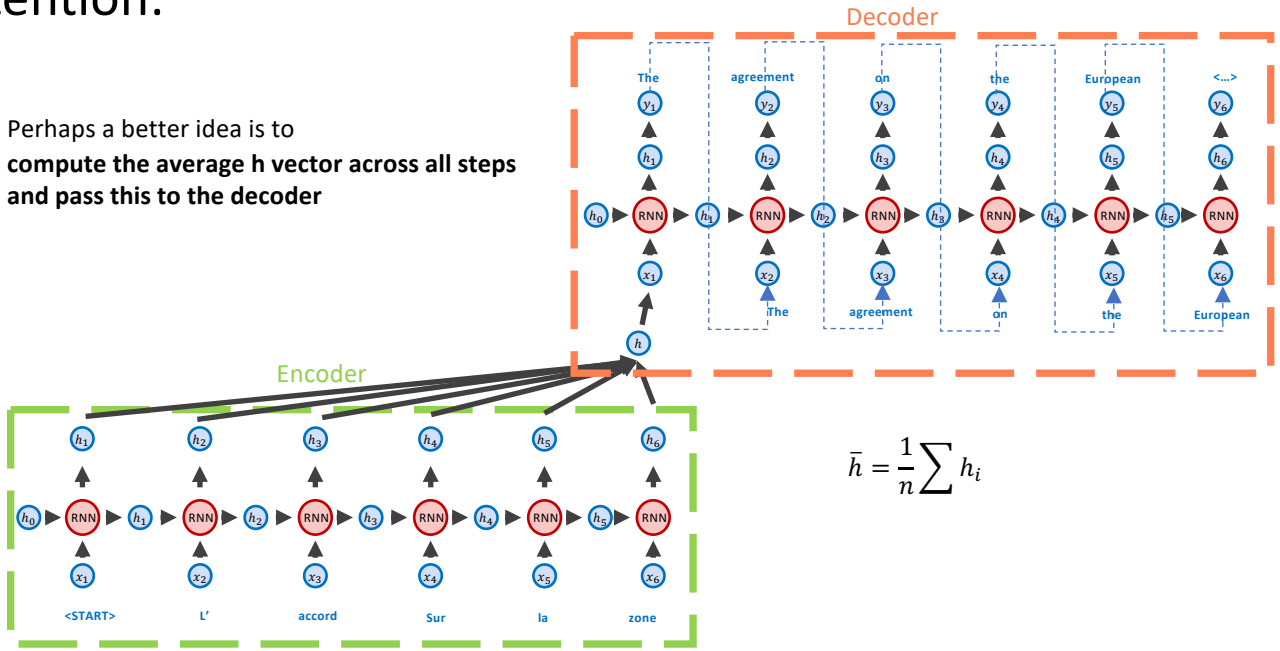


Attention process in NLP

Basic language translation models: Encoder/Decoder

Cross-attention:

Perhaps a better idea is to
compute the average h vector across all steps and pass this to the decoder



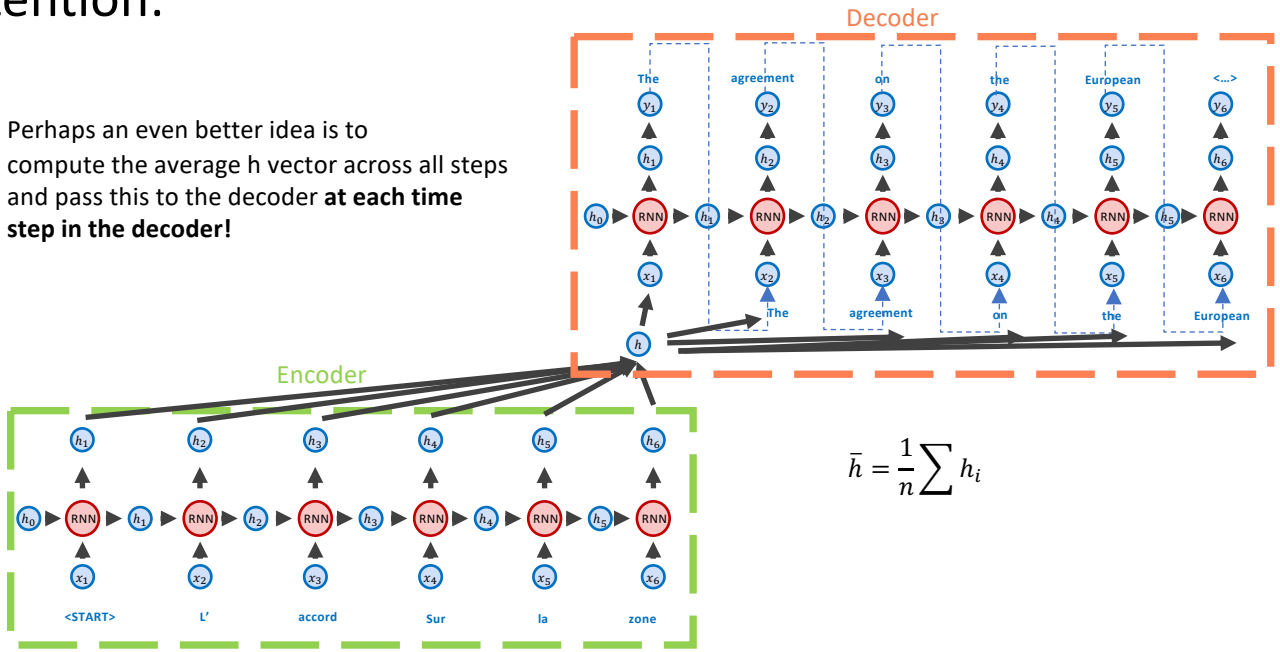
$$\bar{h} = \frac{1}{n}\sum h_i$$

Attention process in NLP

Basic language translation models: Encoder/Decoder

Cross-attention:

Perhaps an even better idea is to compute the average h vector across all steps and pass this to the decoder **at each time step in the decoder!**

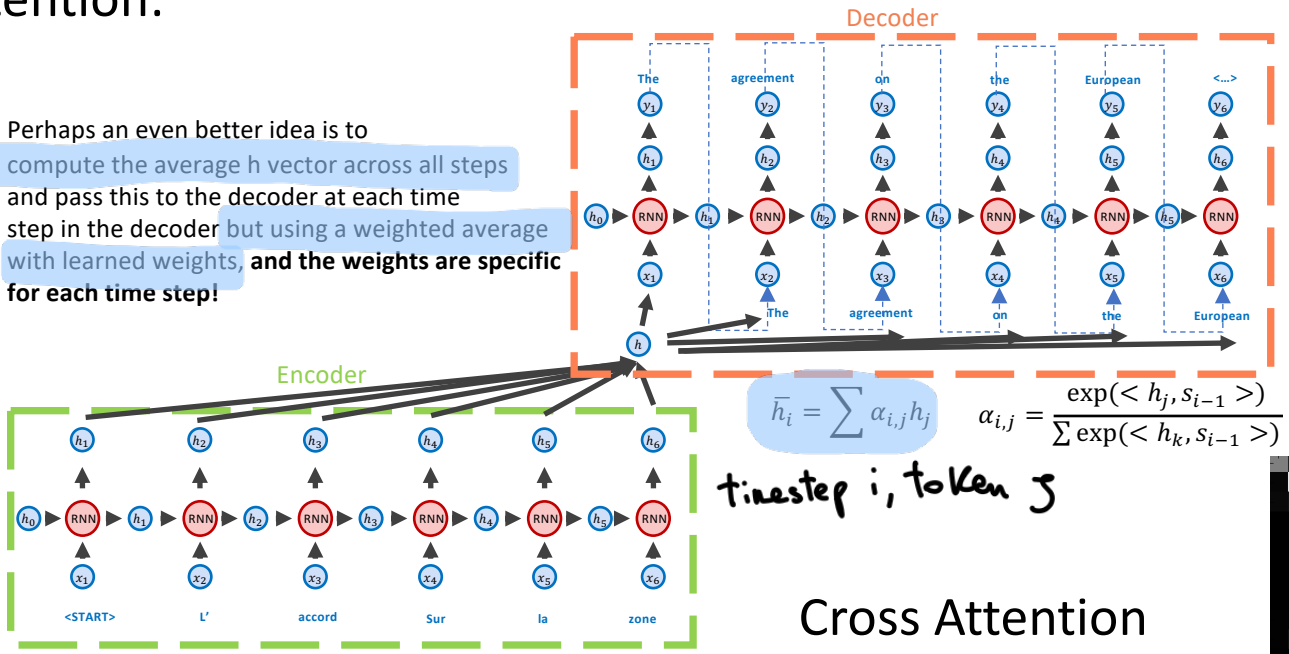


$$\bar{h} = \frac{1}{n}\sum h_i$$

Attention process in NLP

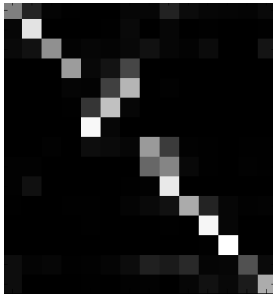
Basic language translation models: Encoder/Decoder

Cross-attention:



Cross Attention

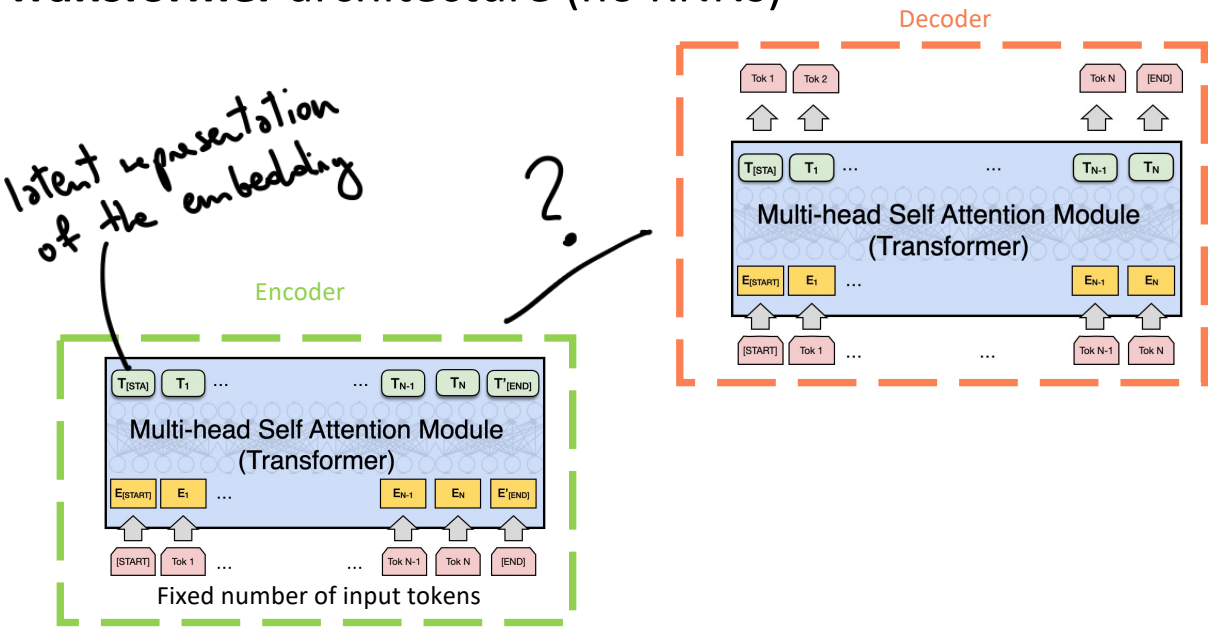
Encoder/ Decoder



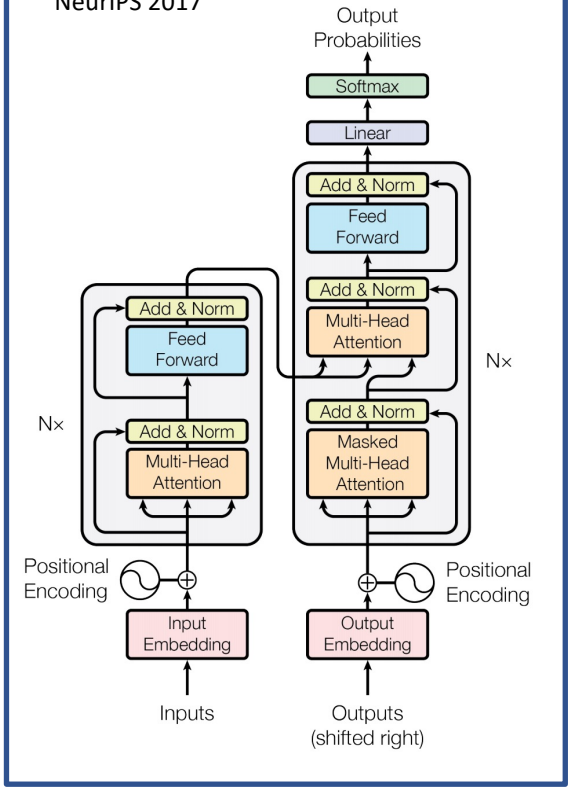
Attention process in NLP

Basic language translation models: Encoder/Decoder

Transformer architecture (no RNNs)



[Vaswani et al. Attention is all you need]
https://arxiv.org/abs/1706.03762
NeurIPS 2017

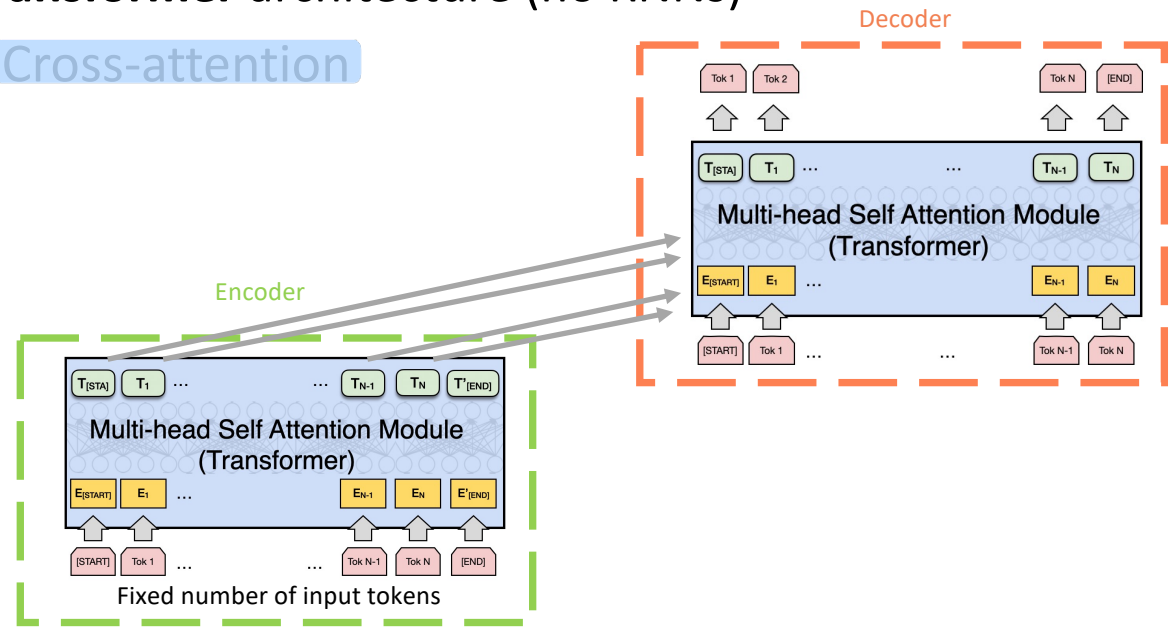


Attention process in NLP

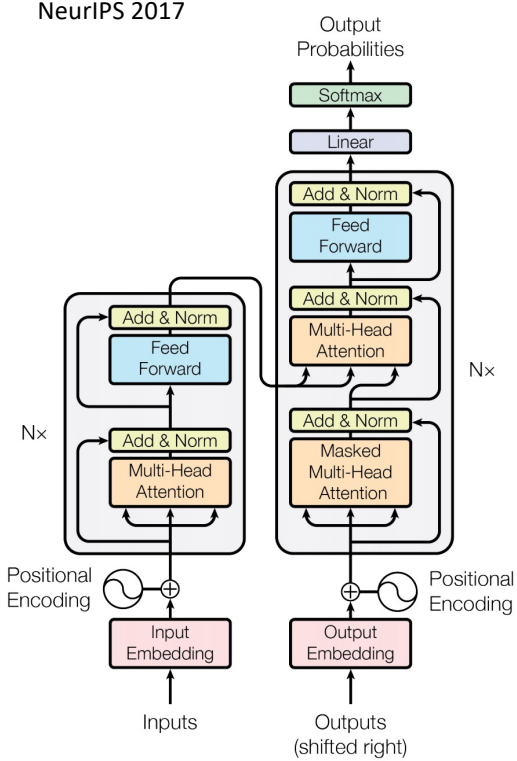
Basic language translation models: Encoder/Decoder

Transformer architecture (no RNNs)

- Cross-attention



[Vaswani et al. Attention is all you need]
https://arxiv.org/abs/1706.03762
NeurIPS 2017

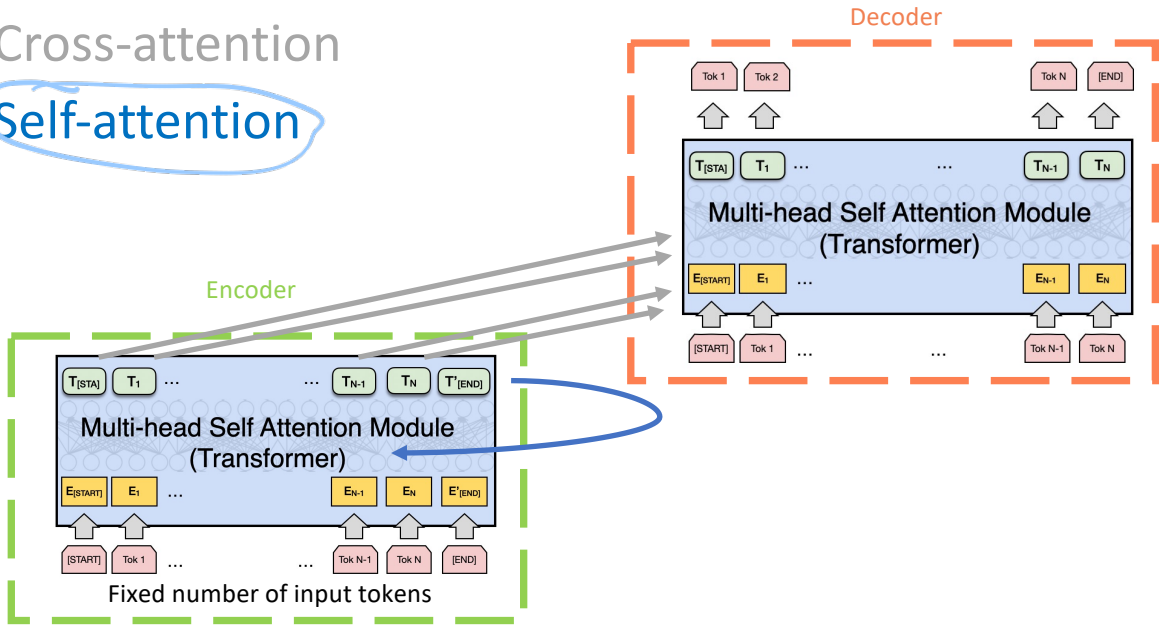


Attention process in NLP

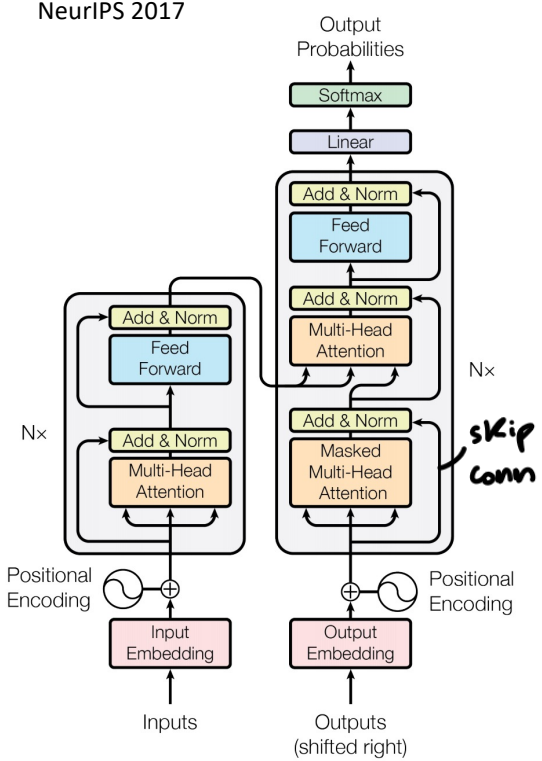
Basic language translation models: Encoder/Decoder

Transformer architecture (no RNNs)

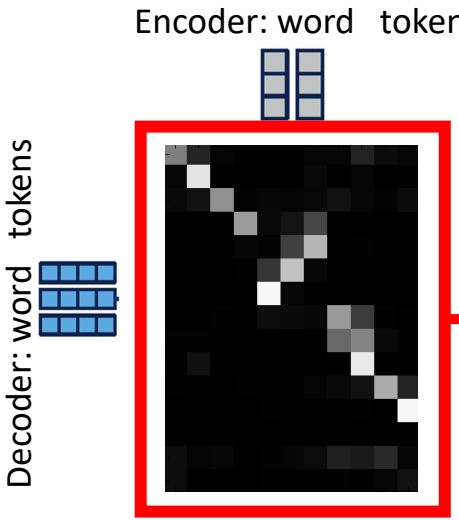
- Cross-attention
- Self-attention



[Vaswani et al. Attention is all you need]
https://arxiv.org/abs/1706.03762
NeurIPS 2017

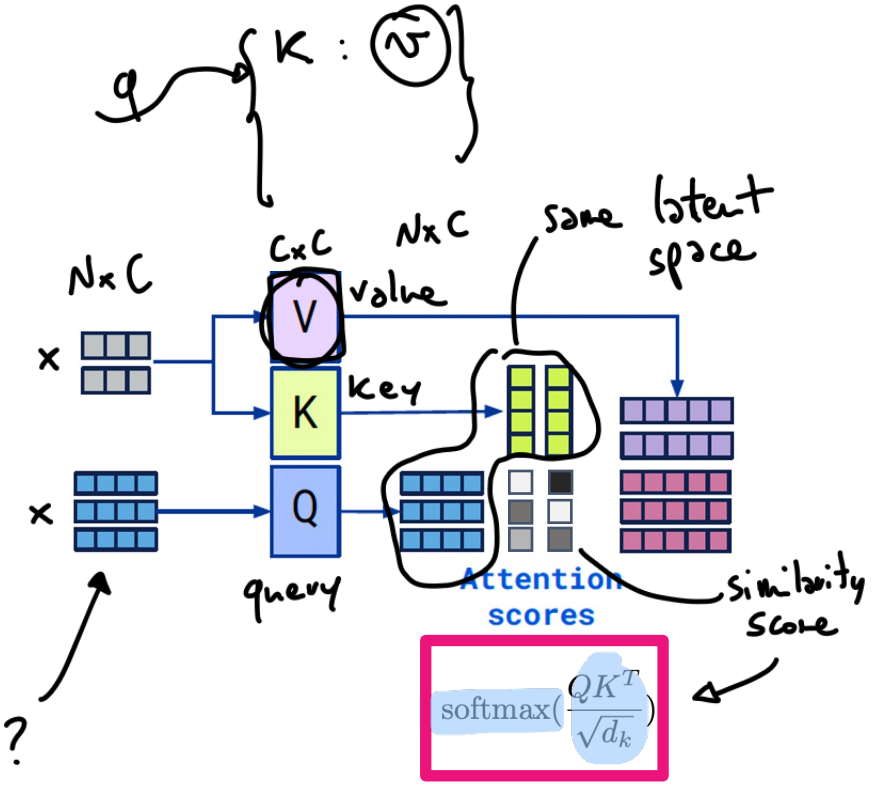


Attention process in NLP



Cross Attention Module

What are the best tokens coming from the decoder?



$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

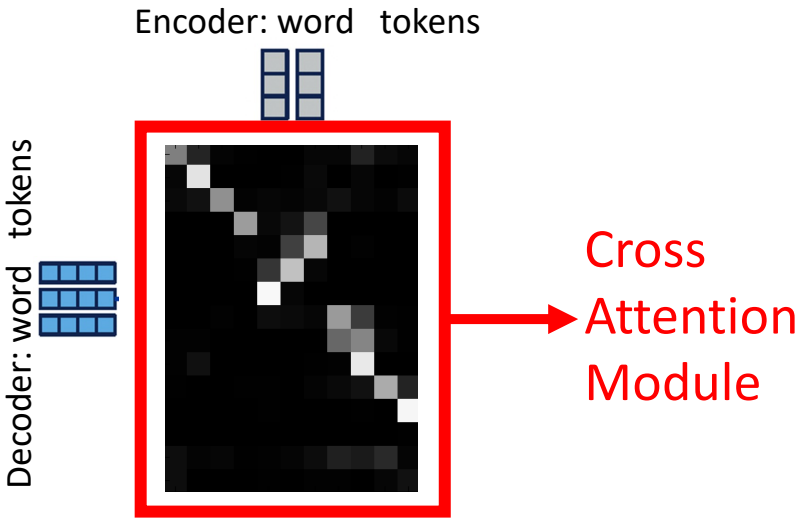
Outline

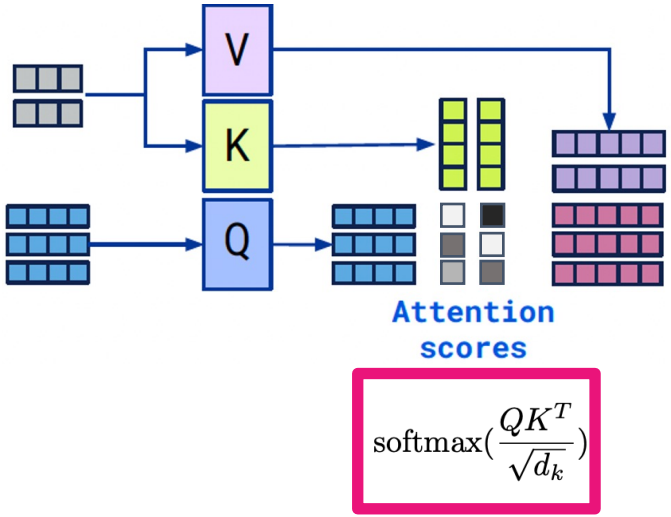
1. Computer Vision and ML basics: Visual (local) feature detection and description, Bag of Word Image representation, Linear classification (SVM)
2. Introduction to Neural Networks (NNs)
3. Machine Learning basics (2): Risk, Classification, Datasets, benchmarks and evaluation
4. Neural Nets for Image Classification
5. Vision Transformers

   NLP: Attention is all you need

   Transformer for image classification

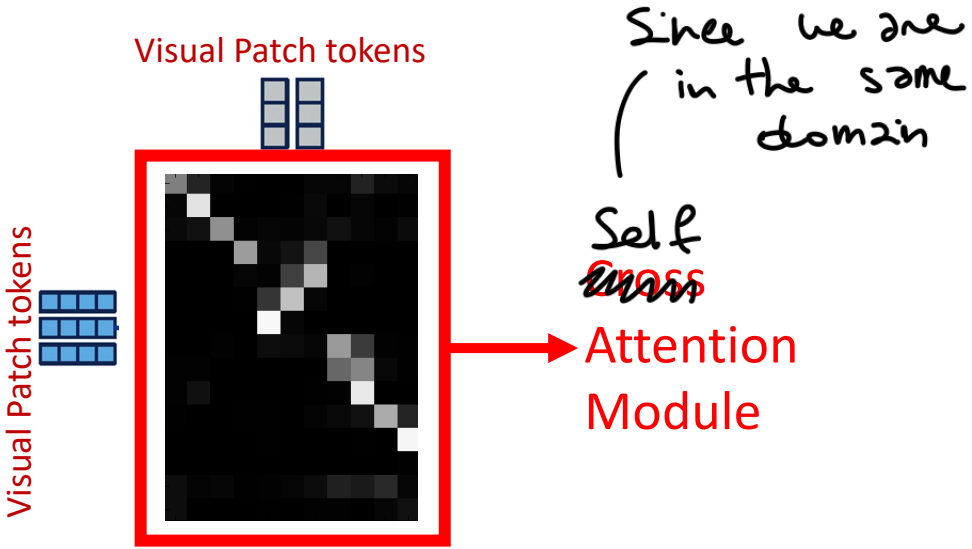
Attention process in NLP

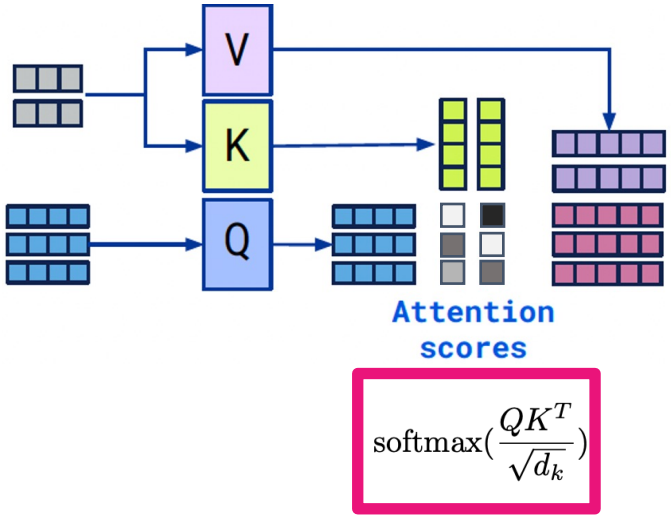




$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

Attention process in Vision





$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

Very similar except that Visual token is definitively less natural than word for NLP

Attention process in Vision

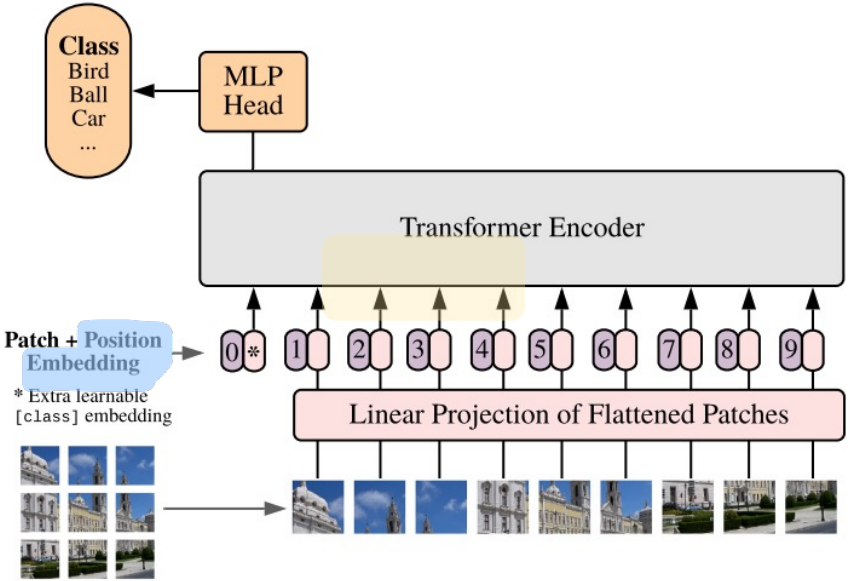
Is it possible to mimic this attention-based architecture for vision processing?

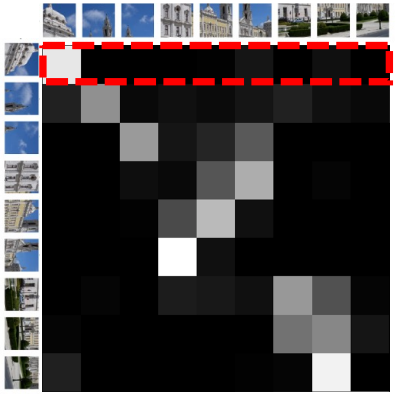
Yes! **ViT** (Vision image Transformers) architecture

Published as a conference paper at ICLR 2021

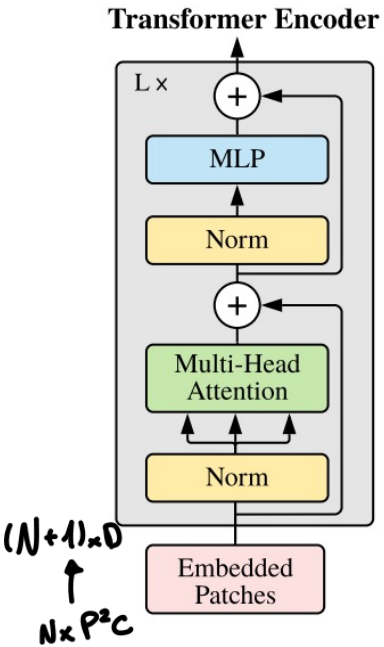
AN IMAGE IS WORTH 16X16 WORDS: TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE

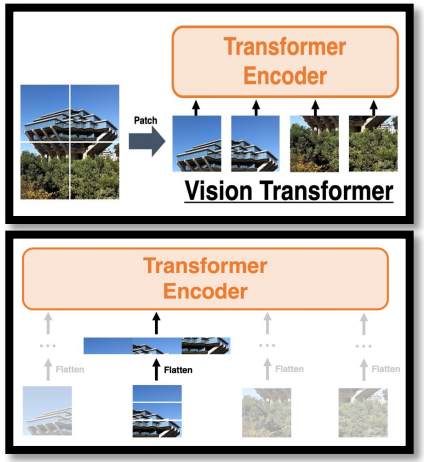
Alexey Dosovitskiy[*,†], **Lucas Beyer**[*], **Alexander Kolesnikov**[*], **Dirk Weissenborn**[*], **Xiaohua Zhai**[*], **Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, Neil Houlsby**[*,†]
[*]equal technical contribution, [†]equal advising
Google Research, Brain Team
{adosovitskiy, neilhoulsby}@google.com



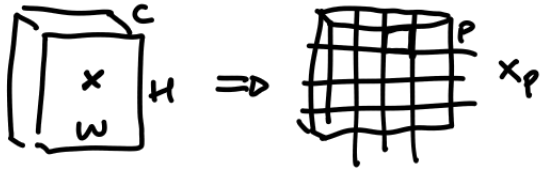


Attention process in Vision





each patch is PxPxC

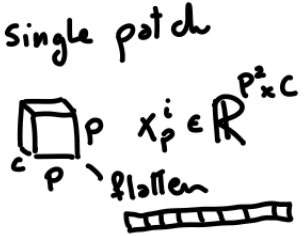


$x \in \mathbb{R}^{H \times W \times C}$

$x_p \in \mathbb{R}^{N \times (P^2 \cdot C)}$

$N = HW/P^2$

Single patch



$$\mathbf{z}_0 = [\mathbf{x}_{\text{class}}; \mathbf{x}_p^1\mathbf{E}; \mathbf{x}_p^2\mathbf{E}; \cdots; \mathbf{x}_p^N\mathbf{E}] + \mathbf{E}_{pos},$$

$$\mathbf{z}'_\ell = \text{MSA}(\text{LN}(\mathbf{z}_{\ell-1})) + \mathbf{z}_{\ell-1}, \quad \ell = 1 \ldots L$$

$$\mathbf{z}_\ell = \text{MLP}(\text{LN}(\mathbf{z}'_\ell)) + \mathbf{z}'_\ell, \quad \ell = 1 \ldots L$$

$$\mathbf{y} = \text{LN}(\mathbf{z}_L^0)$$

CLS token

$$\mathbf{E} \in \mathbb{R}^{(P^2 \cdot C) \times D}, \; \mathbf{E}_{pos} \in \mathbb{R}^{(N+1) \times D}$$

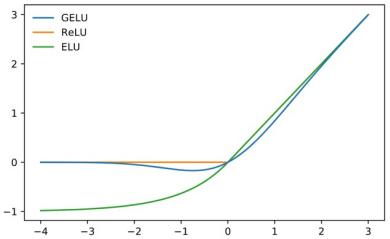
[class=CLS] token: a learnable embedding to the sequence of embedded patches

Layernorm (LN) before every block, and residual connections after every block

MSA: Multi Head Self Attention

MLP: two layers with a GELU non-linearity

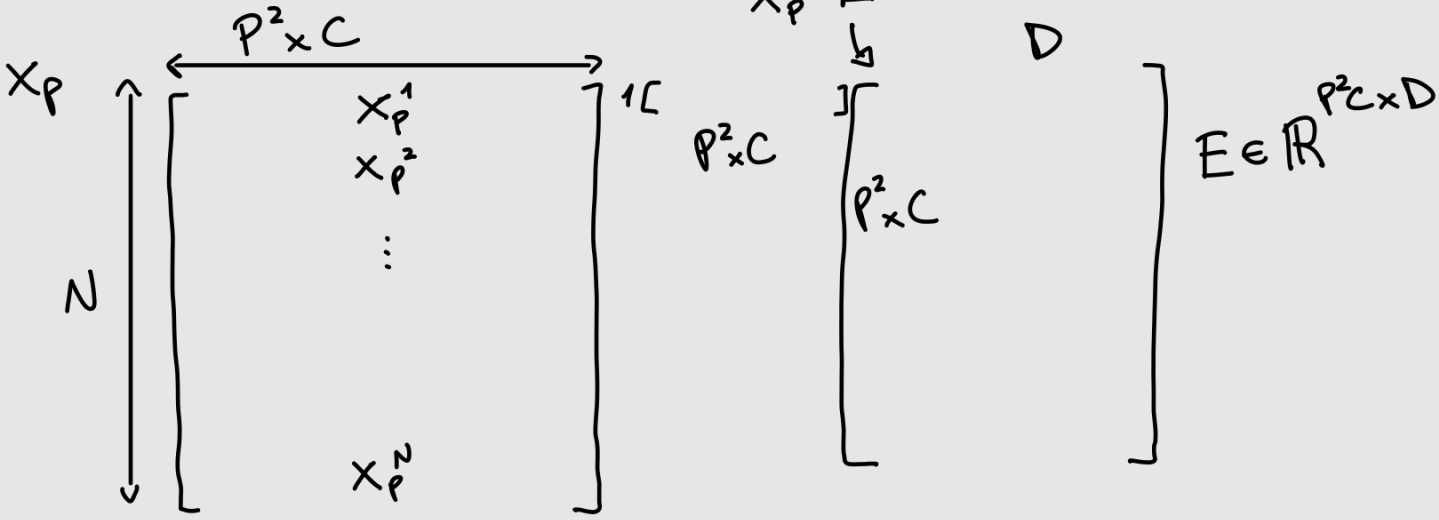
Hybrid Architecture : Raw image patches --> Feature map of a CNN



$x_p \in \mathbb{R}^{N \times (P^2 \times C)}$

$1 \times (P^2 C)(P^2 C) \times D$ — to embed x_p^i in latent dimension $\mathbb{R}^D$

$x_p^i \; E$



$$[\, x_p^i \,] \begin{bmatrix} E \end{bmatrix} = [\, z_0^i \,] \qquad E_{pos} = 0$$

token of the class CLS

$$z_0 \in \mathbb{R}^{(N+1) \times D}$$

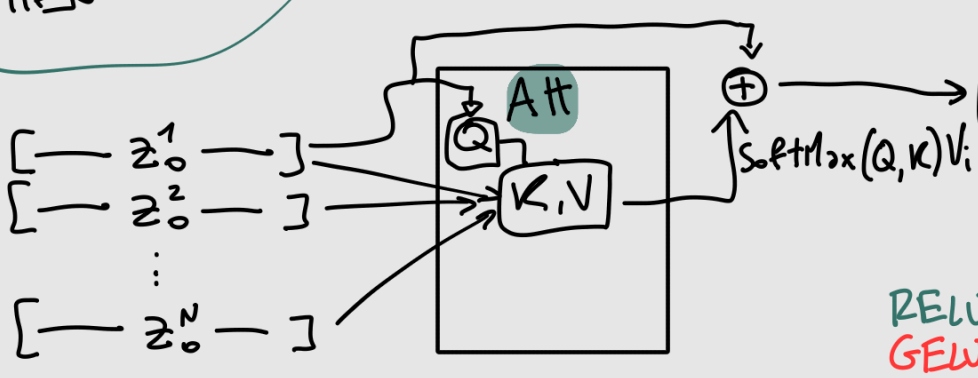
$$\begin{bmatrix} x_{class} \\ z_0^1 \\ \vdots \\ z_0^N \end{bmatrix} \; N+1$$

EMBEDDING + POSITION

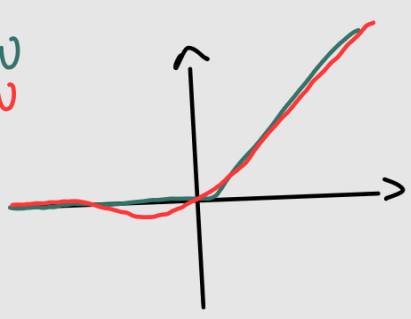
ATTENTION

residual skip

$$(z_0^i)' = SA(LN(z_0^i)) + z_0^i$$



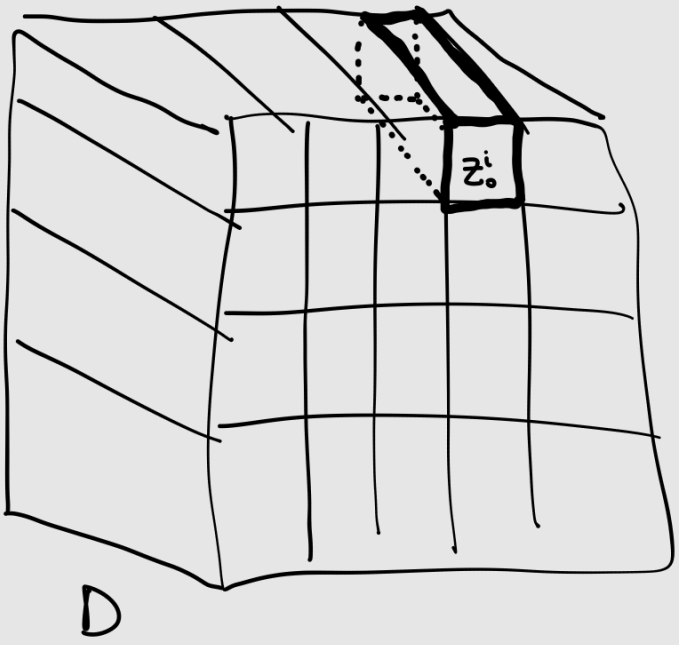
RELU
GELU



$(z_0^i)' \rightarrow LN \rightarrow MLP \rightarrow \oplus \rightarrow (z_1^i)$

2 layers $\quad W_2(GELU(W_1(z_0^i)'))$

MLP



Z_0

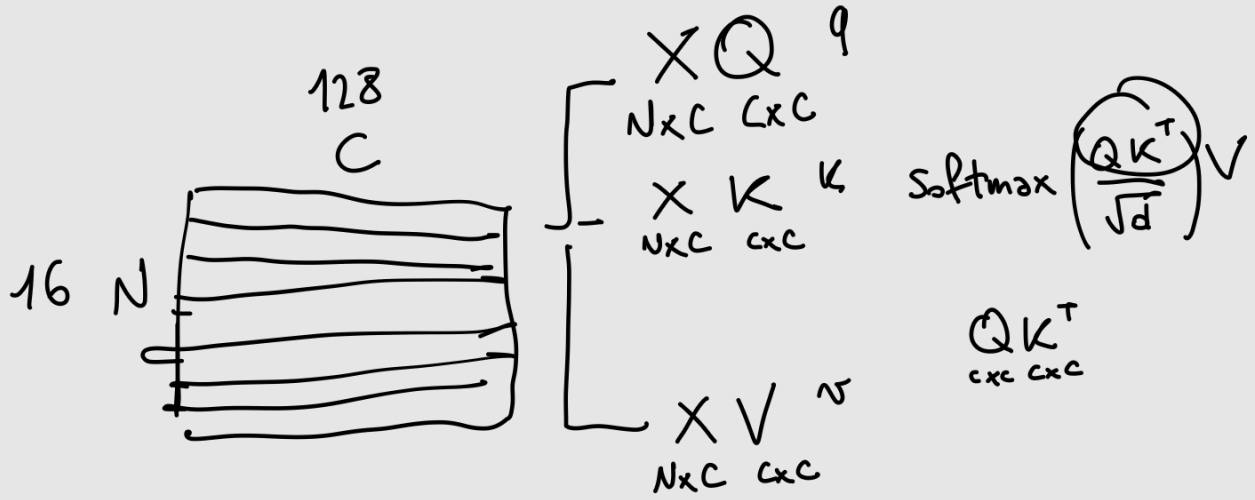
W_1

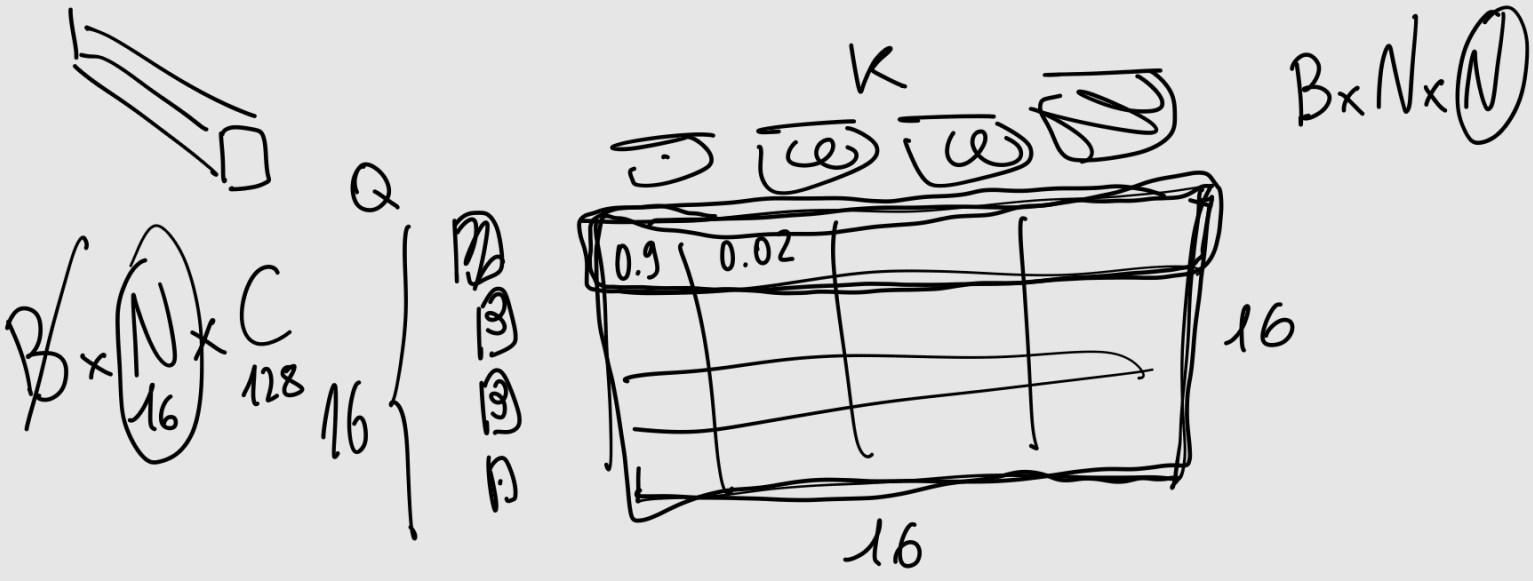
$$\begin{bmatrix} W_{1_1} \\ \vdots \end{bmatrix} \begin{bmatrix} z_0^i \end{bmatrix}$$

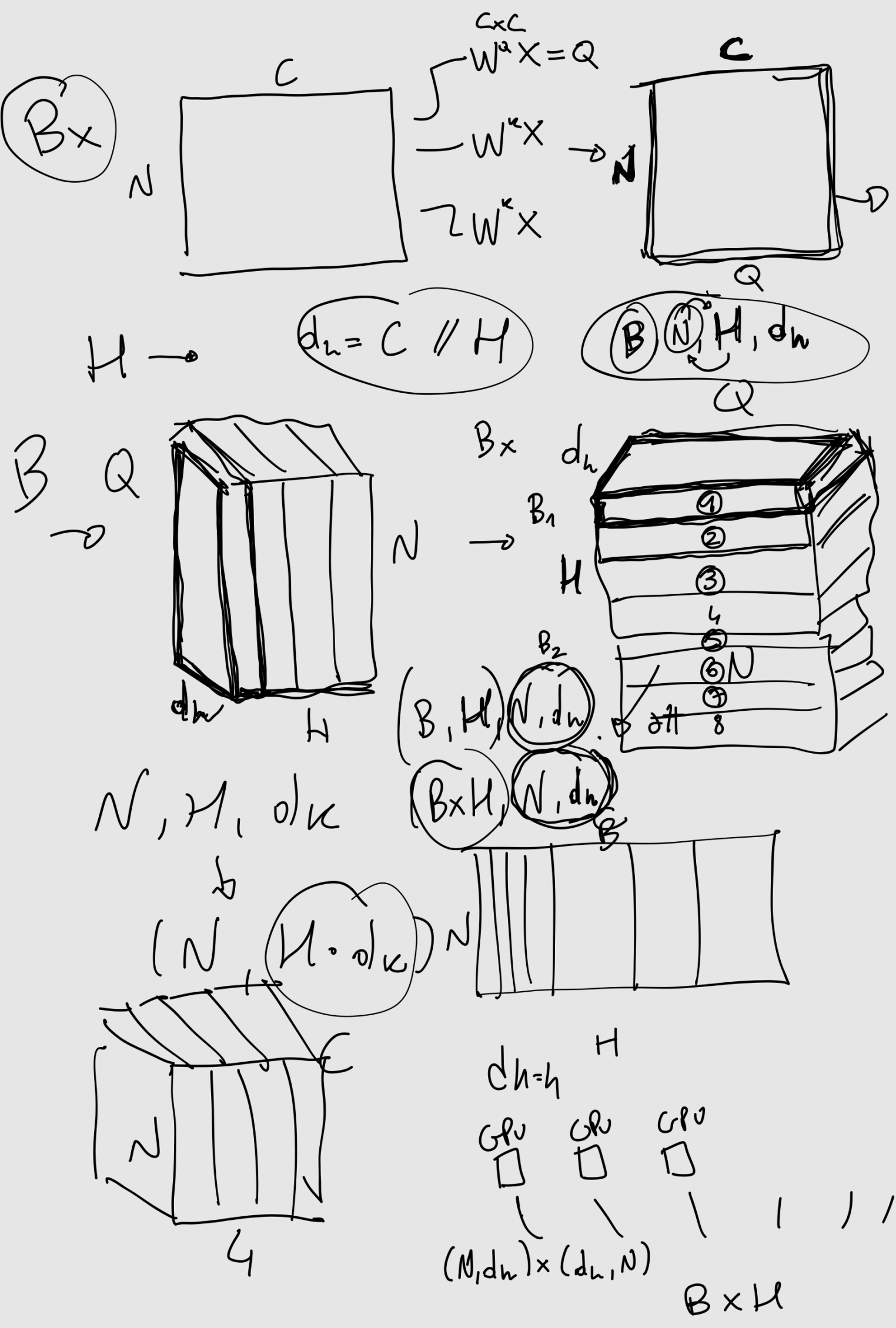
W_{1_J} is

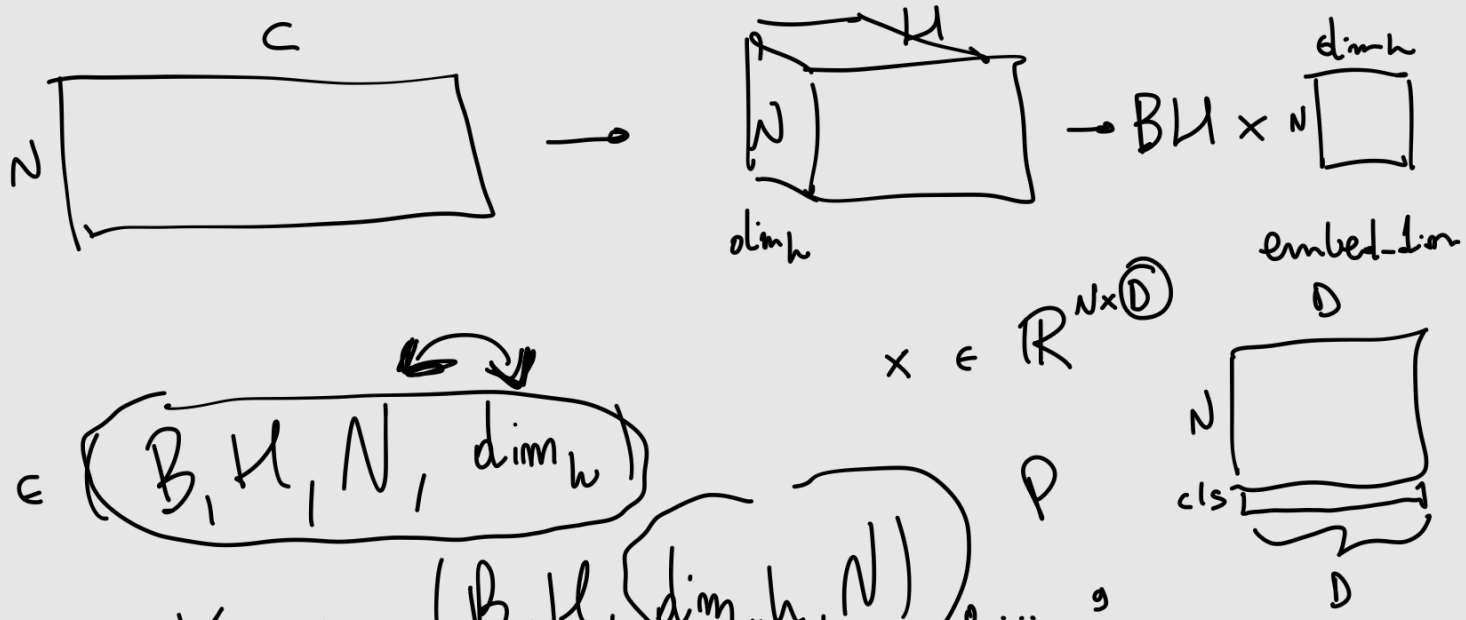
1x1 conv with D filters

$W_1 z_0^i$ has the same dimensions and it can be used as a saliency map to interpret the model






C×C
$W^q X = Q$
$W^k X$
$W^v X$
C
N
N
Q
Bx
H →
$d_h = C // H$
B N H, d_h
Q
B Q
N
d_h
H
Bx
d_h
B_1
H
1
2
3
4
5
6 N
7
8
B_2
B, H, N, d_h
B×H, N, d_h
B
N, H, d_k
(N, H · d_k)
N
H
$d_h = 4$
GPU
GPU
GPU
(N, d_h) × (d_h, N)
B × H

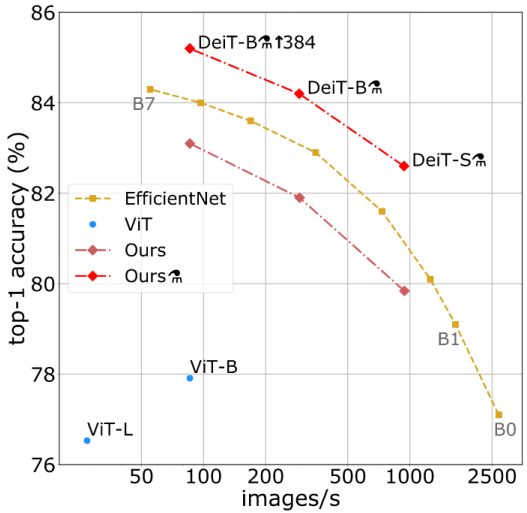

C
N
H
N
dim_h
BH × N
dim_h
embed_dim
D
N
cls
D
x ∈ R^{N×D}
Q ∈ (B, H, N, dim_h)
P
K.view(B, H, dim_h, N)
x = B, H, N, dim_h
x = (B, N, C)
N
N

Attention process in Vision

Experiments with ViT (and variants DeiT, CaiT) transformers for image classification

State-of-the-art performance on ImageNet1k classification!

From ViT paper, **many tricks/discussions to simplify learning** in DeiT, CaiT, …



DeiT

Published as a conference paper at ICML 2021

Training data-efficient image transformers & distillation through attention

Hugo Touvron [1 2] Matthieu Cord [1 2] Matthijs Douze [1]
Francisco Massa [1] Alexandre Sablayrolles [1] Hervé Jégou [1]