Course 4: Deep Learning

2024-04-24

Course 4: Deep Learning

Course 4: Deep Learning

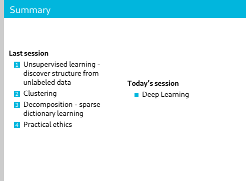
Summary

Last session

1. Unsupervised learning - discover structure from unlabeled data
2. Clustering
3. Decomposition - sparse dictionary learning
4. Practical ethics

Today's session

- Deep Learning



Summary

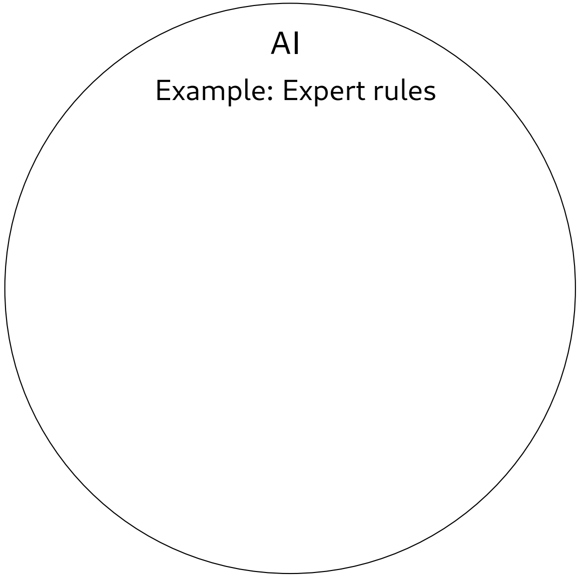
Last session

1. Unsupervised learning - discover structure from unlabeled data
2. Clustering
3. Decomposition - sparse dictionary learning
4. Practical ethics

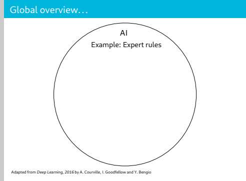
Today's session

- Deep Learning

Global overview…

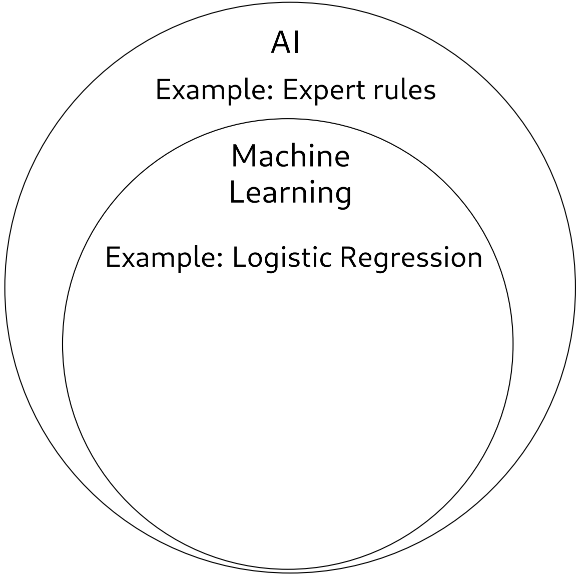


Adapted from *Deep Learning, 2016* by A. Courville, I. Goodfellow and Y. Bengio


Global overview…
AI
Example: Expert rules
Adapted from *Deep Learning, 2016* by A. Courville, I. Goodfellow and Y. Bengio

Deep Learning is a particular case of Machine Learning.

Global overview…



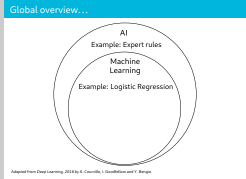
AI

Example: Expert rules

Machine Learning

Example: Logistic Regression

Adapted from *Deep Learning, 2016* by A. Courville, I. Goodfellow and Y. Bengio



Global overview…

AI

Example: Expert rules

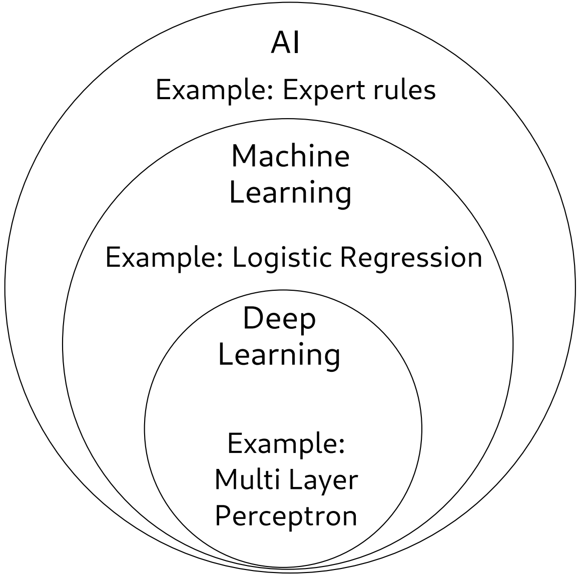
Machine Learning

Example: Logistic Regression

Adapted from Deep Learning, 2016 by A. Courville, I. Goodfellow and Y. Bengio

Deep Learning is a particular case of Machine Learning.

Global overview…



AI

Example: Expert rules

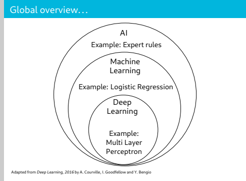
Machine Learning

Example: Logistic Regression

Deep Learning

Example: Multi Layer Perceptron

Adapted from *Deep Learning, 2016* by A. Courville, I. Goodfellow and Y. Bengio



Global overview…

AI
Example: Expert rules
Machine Learning
Example: Logistic Regression
Deep Learning
Example: Multi Layer Perceptron

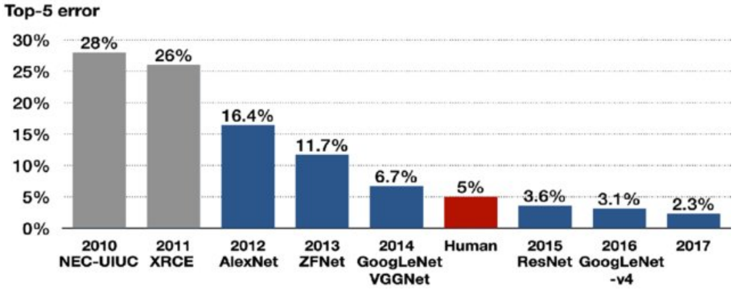
Adapted from *Deep Learning, 2016* by A. Courville, I. Goodfellow and Y. Bengio

Deep Learning is a particular case of Machine Learning.

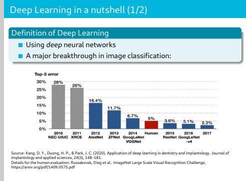
Deep Learning in a nutshell (1/2)

Definition of Deep Learning

- Using *deep* neural networks
- A major breakthrough in image classification:

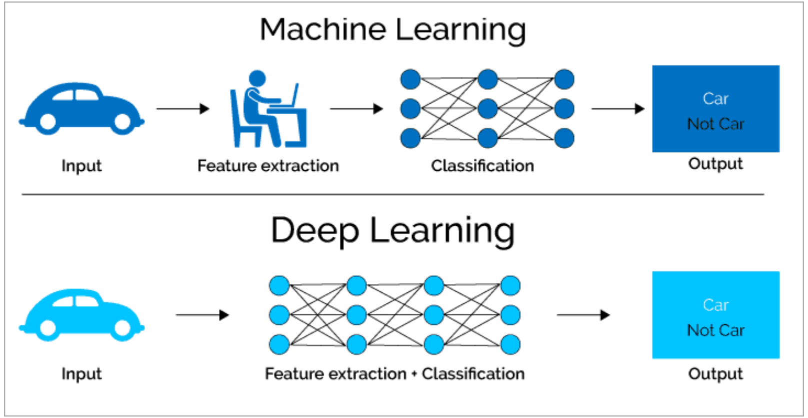


Source: Kang, D. Y., Duong, H. P., & Park, J. C. (2020). Application of deep learning in dentistry and implantology. Journal of implantology and applied sciences, 24(3), 148-181.
Details for the human evaluation: Russakovsk, Dieg *et al.*. ImageNet Large Scale Visual Recognition Challenge, https://arxiv.org/pdf/1409.0575.pdf


Deep Learning in a nutshell (1/2)
Definition of Deep Learning
Using deep neural networks
A major breakthrough in image classification:

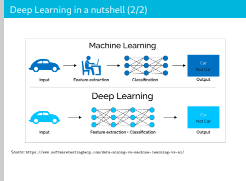
The landscape of Machine Learning changed in 2012: Deep Neural Networks, a technique used in a minority of cases until then, suddenly won the Image Classification contest on ImageNet, a standard image classification dataset. From this point, Deep Neural Networks became mainstream, and the performance of Deep Neural Network models skyrocketed.

Deep Learning in a nutshell (2/2)



Source: `https://www.softwaretestinghelp.com/data-mining-vs-machine-learning-vs-ai/`

└Deep Learning in a nutshell (2/2)

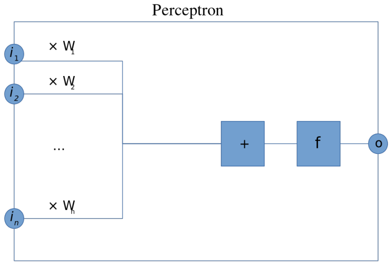

Deep Learning in a nutshell (2/2)

Source: https://www.softwaretestinghelp.com/data-mining-vs-machine-learning-vs-ai/

Deep Neural Networks (1/7)

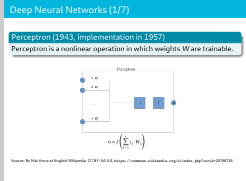
Perceptron (1943, implementation in 1957)

Perceptron is a nonlinear operation in which weights W are trainable.



$$o = f\left(\sum_{k=1}^{n} i_k \cdot W_k\right)$$

Source: By Mat the w at English Wikipedia, CC BY-SA 3.0, `https://commons.wikimedia.org/w/index.php?curid=23766733`


Deep Neural Networks (1/7)
Perceptron (1943, implementation in 1957)
Perceptron is a nonlinear operation in which weights W are trainable.
Source: By Mat the w at English Wikipedia, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=23766733

The Perceptron is a matrix multiplication, followed by a nonlinear function $f()$. It is important to say that the Perceptron is old!! Hence, AI is not a brand new thing, but an old research domain.

Deep Neural Networks (1/7)

Perceptron (1943, implementation in 1957)

Perceptron is a nonlinear operation in which weights *W* are trainable.

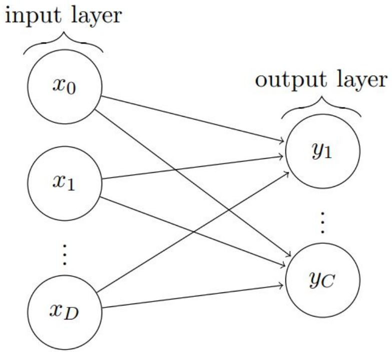
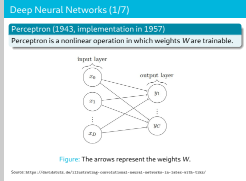


Figure: The arrows represent the weights *W*.

Source: `https://davidstutz.de/illustrating-convolutional-neural-networks-in-latex-with-tikz/`



Deep Neural Networks (1/7)

Perceptron (1943, implementation in 1957)

Perceptron is a nonlinear operation in which weights *W* are trainable.

Figure: The arrows represent the weights *W*.

Source: https://davidstutz.de/illustrating-convolutional-neural-networks-in-latex-with-tikz/

The Perceptron is a matrix multiplication, followed by a nonlinear function *f*(). It is important to say that the Perceptron is old!! Hence, AI is not a brand new thing, but an old research domain.

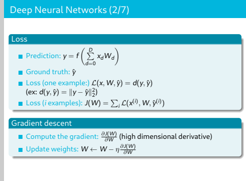
Deep Neural Networks (2/7)

Loss

- Prediction: $y = f\left(\sum_{d=0}^{D} x_d W_d\right)$
- Ground truth: $\hat{y}$
- Loss (one example:) $\mathcal{L}(x, W, \hat{y}) = d(y, \hat{y})$
  (ex: $d(y, \hat{y}) = \|y - \hat{y}\|_2^2$)
- Loss (i examples): $J(W) = \sum_i \mathcal{L}(x^{(i)}, W, \hat{y}^{(i)})$

Gradient descent

- Compute the gradient: $\frac{\partial J(W)}{\partial W}$ (high dimensional derivative)
- Update weights: $W \leftarrow W - \eta \frac{\partial J(W)}{\partial W}$



Deep Neural Networks (2/7)

Loss

- Prediction: $y = f\left(\sum_{d=0}^{D} x_d W_d\right)$
- Ground truth: $\hat{y}$
- Loss (one example:) $\mathcal{L}(x, W, \hat{y}) = d(y, \hat{y})$
  (ex: $d(y, \hat{y}) = \|y - \hat{y}\|_2^2$)
- Loss (i examples): $J(W) = \sum_i \mathcal{L}(x^{(i)}, W, \hat{y}^{(i)})$

Gradient descent

- Compute the gradient: $\frac{\partial J(W)}{\partial W}$ (high dimensional derivative)
- Update weights: $W \leftarrow W - \eta \frac{\partial J(W)}{\partial W}$

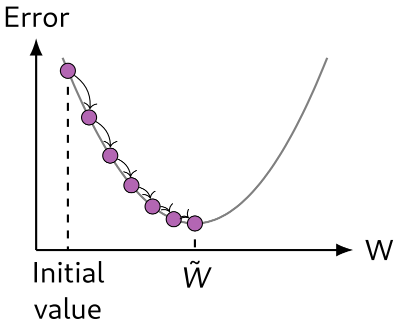
The Perceptron is a matrix multiplication, followed by a nonlinear function $f()$. It is important to say that the Perceptron is old!! Hence, AI is not a brand new thing, but an old research domain.

Deep Neural Networks (3/7)

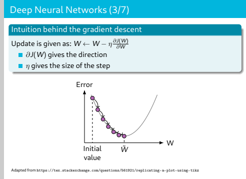
Intuition behind the gradient descent

Update is given as: $W \leftarrow W - \eta\frac{\partial J(W)}{\partial W}$

- $\partial J(W)$ gives the direction
- η gives the size of the step



Adapted from `https://tex.stackexchange.com/questions/561921/replicating-a-plot-using-tikz`


Deep Neural Networks (3/7)
Intuition behind the gradient descent
Update is given as: $W \leftarrow W - \eta\frac{\partial J(W)}{\partial W}$
- $\partial J(W)$ gives the direction
- η gives the size of the step
Adapted from https://tex.stackexchange.com/questions/561921/replicating-a-plot-using-tikz

The gradient follows the increase of the error function, hence the inverse of the gradient follows its decrease. Parameter η, the learning rate, gives the size of the step to take at each iteration. If too small, the model will slowly converge. If too large, the model can be unstable and never reach the optimal solution. In practice, setting the learning rate is not easy.

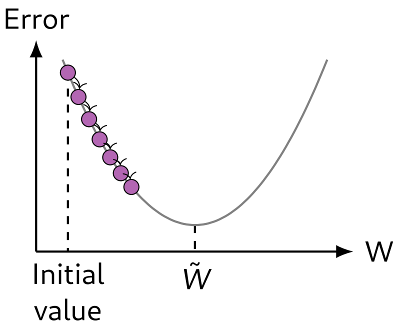
Deep Neural Networks (3/7)

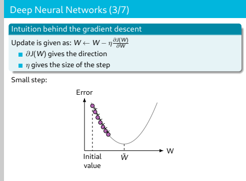
Intuition behind the gradient descent

Update is given as: $W \leftarrow W - \eta \frac{\partial J(W)}{\partial W}$

- $\partial J(W)$ gives the direction
- η gives the size of the step

Small step:




Deep Neural Networks (3/7)
Intuition behind the gradient descent
Update is given as: $W \leftarrow W - \eta \frac{\partial J(W)}{\partial W}$
$\partial J(W)$ gives the direction
η gives the size of the step
Small step:

The gradient follows the increase of the error function, hence the inverse of the gradient follows its decrease. Parameter η, the learning rate, gives the size of the step to take at each iteration. If too small, the model will slowly converge. If too large, the model can be unstable and never reach the optimal solution. In practice, setting the learning rate is not easy.

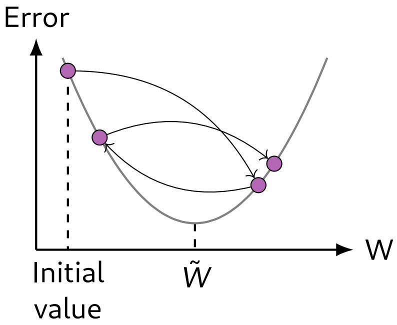
Deep Neural Networks (3/7)

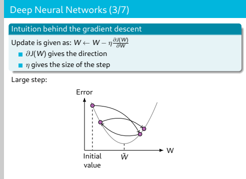
Intuition behind the gradient descent

Update is given as: $W \leftarrow W - \eta \frac{\partial J(W)}{\partial W}$

- $\partial J(W)$ gives the direction
- η gives the size of the step

Large step:



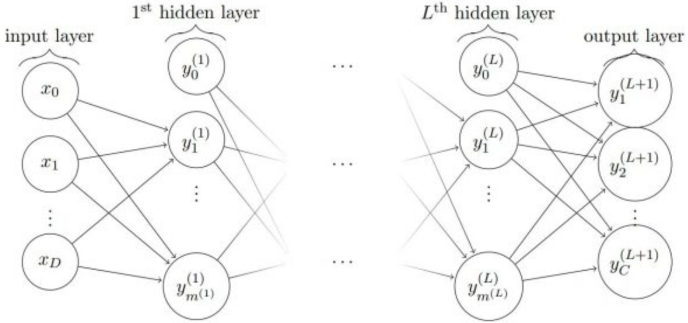

Deep Neural Networks (3/7)
Intuition behind the gradient descent
Update is given as: $W \leftarrow W - \eta \frac{\partial J(W)}{\partial W}$
$\partial J(W)$ gives the direction
η gives the size of the step
Large step:

The gradient follows the increase of the error function, hence the inverse of the gradient follows its decrease. Parameter η, the learning rate, gives the size of the step to take at each iteration. If too small, the model will slowly converge. If too large, the model can be unstable and never reach the optimal solution. In practice, setting the learning rate is not easy.

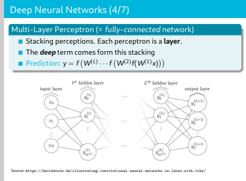
Deep Neural Networks (4/7)

Multi-Layer Perceptron (= *fully-connected* network)

- Stacking perceptions. Each perceptron is a **layer**.
- The ***deep*** term comes form this stacking
- Prediction: $y = f\left(W^{(L)} \cdots f\left(W^{(2)} f(W^{(1)} x)\right)\right)$



Source: `https://davidstutz.de/illustrating-convolutional-neural-networks-in-latex-with-tikz/`


Deep Neural Networks (4/7)
Multi-Layer Perceptron (= *fully-connected* network)
- Stacking perceptions. Each perceptron is a **layer**.
- The ***deep*** term comes form this stacking
- Prediction: $y = f\left(W^{(L)} \cdots f\left(W^{(2)} f(W^{(1)} x)\right)\right)$

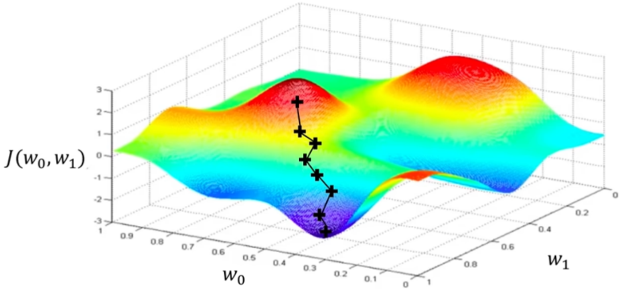
Source: https://davidstutz.de/illustrating-convolutional-neural-networks-in-latex-with-tikz/

The bias are removed from the equations for simplicity, but say that they exist orally.

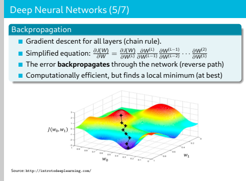
Deep Neural Networks (5/7)

Backpropagation

- Gradient descent for all layers (chain rule).
- Simplified equation: $\frac{\partial J(W)}{\partial W} = \frac{\partial J(W)}{\partial W^{(L)}} \frac{\partial W^{(L)}}{\partial W^{(L-1)}} \frac{\partial W^{(L-1)}}{\partial W^{(L-2)}} \cdots \frac{\partial W^{(2)}}{\partial W^{(1)}}$
- The error **backpropagates** through the network (reverse path)
- Computationally efficient, but finds a local minimum (at best)



Source: `http://introtodeeplearning.com/`

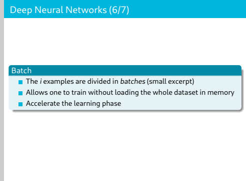

Deep Neural Networks (5/7)
Backpropagation
- Gradient descent for all layers (chain rule).
- Simplified equation: $\frac{\partial J(W)}{\partial W} = \frac{\partial J(W)}{\partial W^{(L)}} \frac{\partial W^{(L)}}{\partial W^{(L-1)}} \frac{\partial W^{(L-1)}}{\partial W^{(L-2)}} \cdots \frac{\partial W^{(2)}}{\partial W^{(1)}}$
- The error **backpropagates** through the network (reverse path)
- Computationally efficient, but finds a local minimum (at best)
Source: http://introtodeeplearning.com/

Careful: the given equation is not correct, just conveniently simplified. For better details, refer to `https://en.wikipedia.org/wiki/Backpropagation`.

Deep Neural Networks (6/7)

Batch

- The *i* examples are divided in *batches* (small excerpt)
- Allows one to train without loading the whole dataset in memory
- Accelerate the learning phase



Deep Neural Networks (6/7)

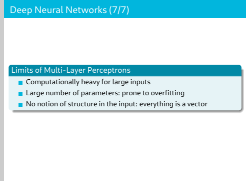
Batch

- The *i* examples are divided in *batches* (small excerpt)
- Allows one to train without loading the whole dataset in memory
- Accelerate the learning phase

Deep Neural Networks (7/7)

Limits of Multi-Layer Perceptrons

- Computationally heavy for large inputs
- Large number of parameters: prone to overfitting
- No notion of structure in the input: everything is a vector



Deep Neural Networks (7/7)

Limits of Multi-Layer Perceptrons

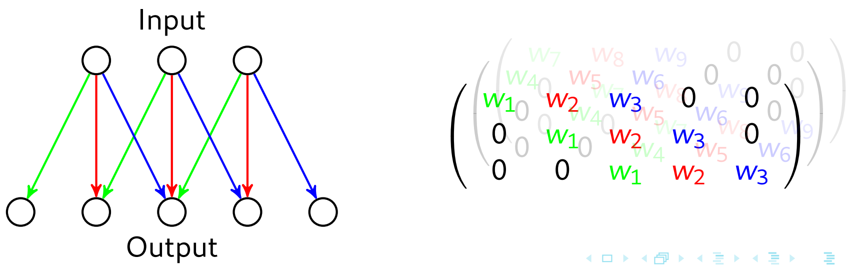
- Computationally heavy for large inputs
- Large number of parameters: prone to overfitting
- No notion of structure in the input: everything is a vector

Convolutional Neural Network (1/5)

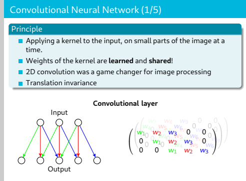
Principle

- Applying a kernel to the input, on small parts of the image at a time.
- Weights of the kernel are **learned** and **shared**!
- 2D convolution was a game changer for image processing
- Translation invariance

Convolutional layer



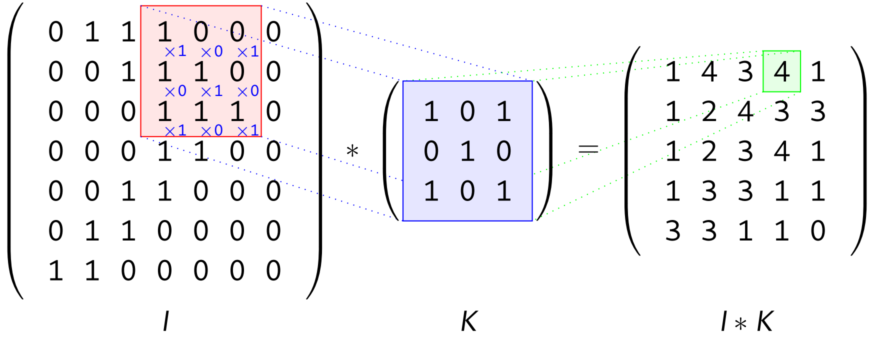
$$\begin{pmatrix} w_1 & w_2 & w_3 & 0 & 0 \\ 0 & w_1 & w_2 & w_3 & 0 \\ 0 & 0 & w_1 & w_2 & w_3 \end{pmatrix}$$


Convolutional Neural Network (1/5)
Principle
- Applying a kernel to the input, on small parts of the image at a time.
- Weights of the kernel are **learned** and **shared**!
- 2D convolution was a game changer for image processing
- Translation invariance
Convolutional layer

Convolution neural networks are the most common architecture for neural networks nowadays. They are particularly successful for image processing, were not challenged until very recently, with Vision Transformers (see Transformers in the last slides). The translation invariance is really important for image processing, as objects can be anywhere on the image. It contributed to the success of Convolutional Neural Networks.

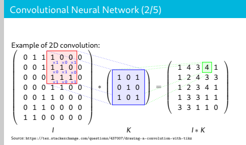
Convolutional Neural Network (2/5)

Example of 2D convolution:



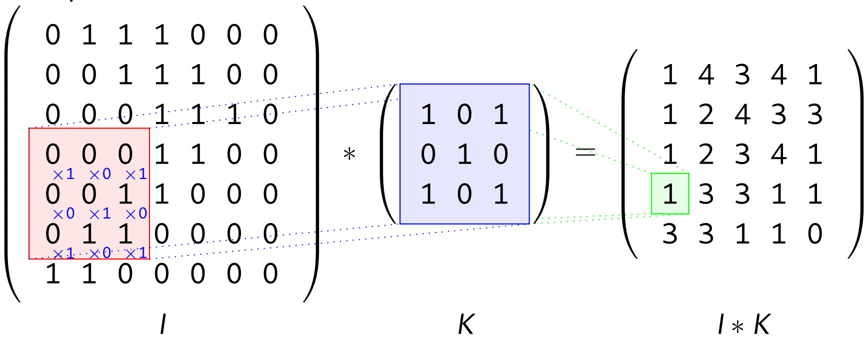
$$\underset{I}{\begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}} * \underset{K}{\begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}} = \underset{I * K}{\begin{pmatrix} 1 & 4 & 3 & \boxed{4} & 1 \\ 1 & 2 & 4 & 3 & 3 \\ 1 & 2 & 3 & 4 & 1 \\ 1 & 3 & 3 & 1 & 1 \\ 3 & 3 & 1 & 1 & 0 \end{pmatrix}}$$

Source: `https://tex.stackexchange.com/questions/437007/drawing-a-convolution-with-tikz`



Convolutional Neural Network (2/5)

Example of 2D convolution:



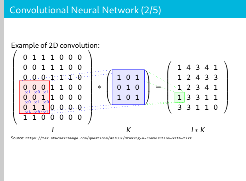
$$\begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} * \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 4 & 3 & 4 & 1 \\ 1 & 2 & 4 & 3 & 3 \\ 1 & 2 & 3 & 4 & 1 \\ 1 & 3 & 3 & 1 & 1 \\ 3 & 3 & 1 & 1 & 0 \end{pmatrix}$$

Source: `https://tex.stackexchange.com/questions/437007/drawing-a-convolution-with-tikz`

2024-04-24

Course 4: Deep Learning

└Convolutional Neural Network (2/5)



Convolutional Neural Network (2/5)

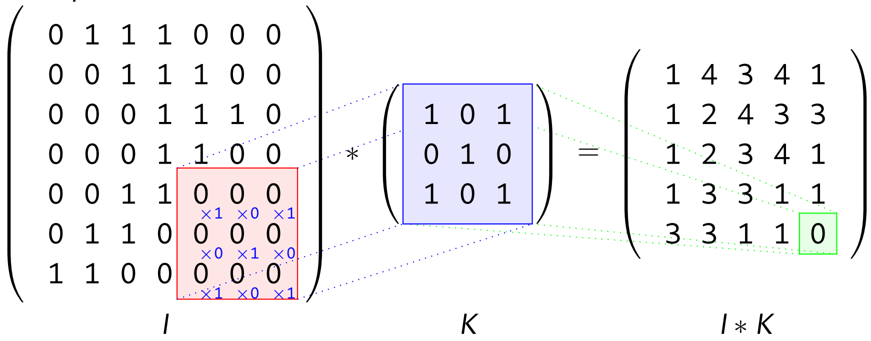
Example of 2D convolution:

I K $I * K$

Source: https://tex.stackexchange.com/questions/437007/drawing-a-convolution-with-tikz

Convolutional Neural Network (2/5)

Example of 2D convolution:



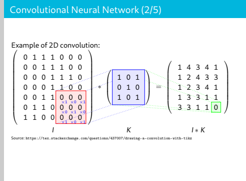
$$\underset{I}{\begin{pmatrix} 0&1&1&1&0&0&0\\ 0&0&1&1&1&0&0\\ 0&0&0&1&1&1&0\\ 0&0&0&1&1&0&0\\ 0&0&1&1&\underset{\times 1}{0}&\underset{\times 0}{0}&\underset{\times 1}{0}\\ 0&1&1&0&\underset{\times 0}{0}&\underset{\times 1}{0}&\underset{\times 0}{0}\\ 1&1&0&0&\underset{\times 1}{0}&\underset{\times 0}{0}&\underset{\times 1}{0} \end{pmatrix}} * \underset{K}{\begin{pmatrix} 1&0&1\\ 0&1&0\\ 1&0&1 \end{pmatrix}} = \underset{I * K}{\begin{pmatrix} 1&4&3&4&1\\ 1&2&4&3&3\\ 1&2&3&4&1\\ 1&3&3&1&1\\ 3&3&1&1&\boxed{0} \end{pmatrix}}$$

Source: `https://tex.stackexchange.com/questions/437007/drawing-a-convolution-with-tikz`

2024-04-24

Course 4: Deep Learning

└Convolutional Neural Network (2/5)

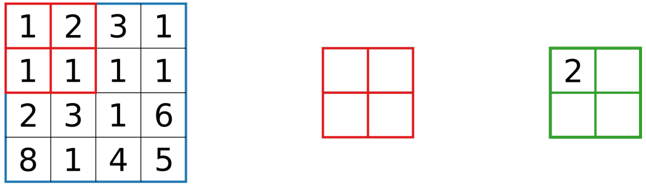


Convolutional Neural Network (2/5)

Example of 2D convolution:

Convolutional Neural Network (3/5)

Example of 2D pooling:



1	2	3	1
1	1	1	1
2	3	1	6
8	1	4	5

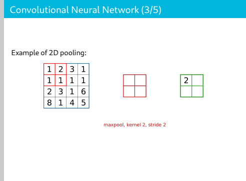
2	

maxpool, kernel 2, stride 2

2024-04-24

Course 4: Deep Learning

└Convolutional Neural Network (3/5)



Convolutional Neural Network (3/5)

Example of 2D pooling:

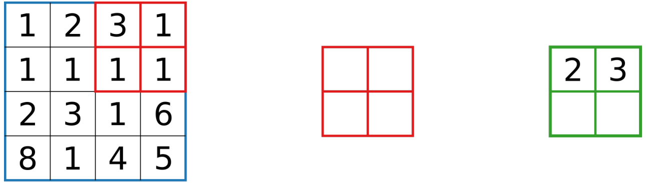
1	2	3	1
1	1	1	1
2	3	1	6
8	1	4	5

2	

maxpool, kernel 2, stride 2

Convolutional Neural Network (3/5)

Example of 2D pooling:



1	2	3	1
1	1	1	1
2	3	1	6
8	1	4	5

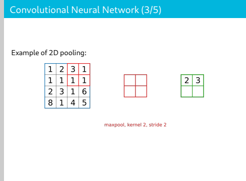
2	3

maxpool, kernel 2, stride 2

2024-04-24

Course 4: Deep Learning

└Convolutional Neural Network (3/5)



Convolutional Neural Network (3/5)

Example of 2D pooling:

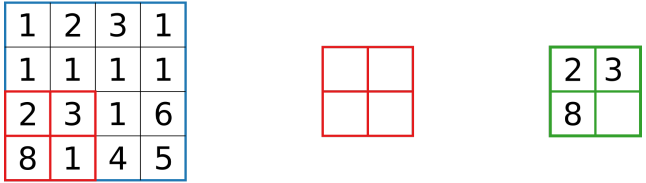
1	2	3	1
1	1	1	1
2	3	1	6
8	1	4	5

2	3

maxpool, kernel 2, stride 2

Convolutional Neural Network (3/5)

Example of 2D pooling:



1	2	3	1
1	1	1	1
2	3	1	6
8	1	4	5

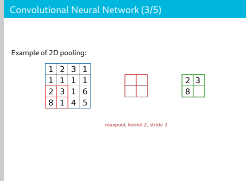
2	3
8	

maxpool, kernel 2, stride 2

2024-04-24

Course 4: Deep Learning

└Convolutional Neural Network (3/5)



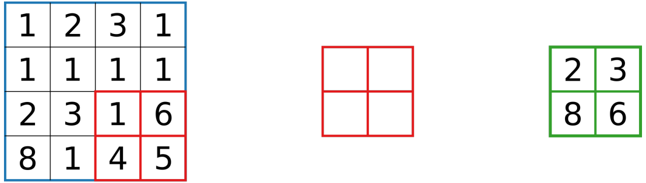
Convolutional Neural Network (3/5)

Example of 2D pooling:

maxpool, kernel 2, stride 2

Convolutional Neural Network (3/5)

Example of 2D pooling:



1	2	3	1
1	1	1	1
2	3	1	6
8	1	4	5

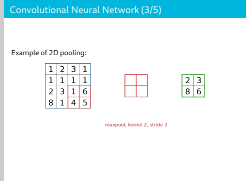
2	3
8	6

maxpool, kernel 2, stride 2

2024-04-24

Course 4: Deep Learning

└Convolutional Neural Network (3/5)



Convolutional Neural Network (3/5)

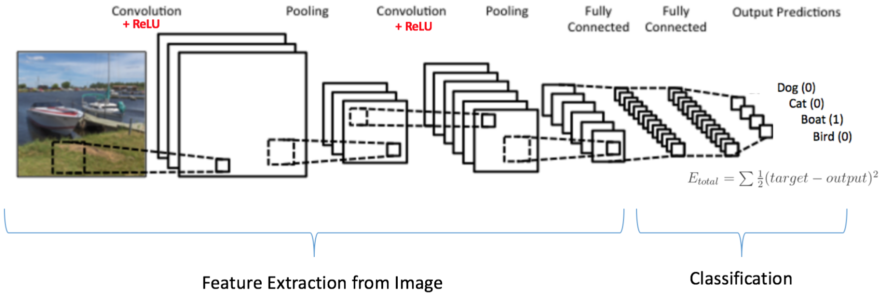
Example of 2D pooling:

maxpool, kernel 2, stride 2

Convolutional Neural Network (4/5)

And repeat...

- Convolutional neural network: mainly *Convolution* + *Pooling*.
- ...But many other components may be added! (batch norm, dropout, skip connections, concatenation, ...)

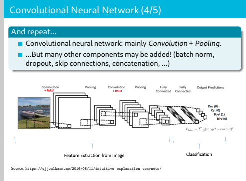


Source: `https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/`

2024-04-24

Course 4: Deep Learning

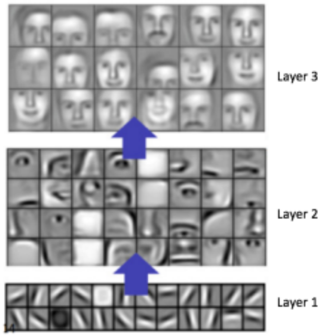
└Convolutional Neural Network (4/5)



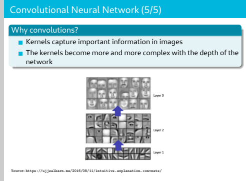
Convolutional Neural Network (5/5)

Why convolutions?

- Kernels capture important information in images
- The kernels become more and more complex with the depth of the network

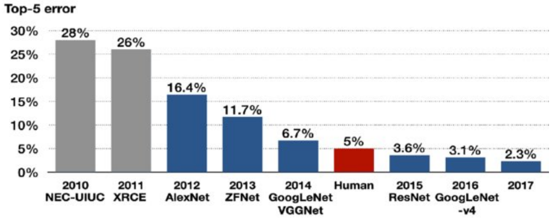


Source: `https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/`


Convolutional Neural Network (5/5)
Why convolutions?
- Kernels capture important information in images
- The kernels become more and more complex with the depth of the network
Source: https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/

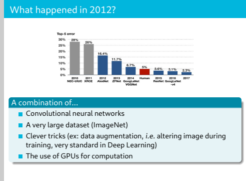
Convolution are able to catch simple shapes (lines, edges) which turn into complex shapes in the subsequent layers.

What happened in 2012?



A combination of...

- Convolutional neural networks
- A very large dataset (ImageNet)
- Clever tricks (ex: data augmentation, *i.e.* altering image during training, very standard in Deep Learning)
- The use of GPUs for computation


What happened in 2012?

A combination of...

- Convolutional neural networks
- A very large dataset (ImageNet)
- Clever tricks (ex: data augmentation, *i.e.* altering image during training, very standard in Deep Learning)
- The use of GPUs for computation

What about now?

Image classification

- Image classification for a single dataset is (almost) solved
- Challenges of adapting models to unseen datasets
- Challenges when data is scarce
- Specific domains with few variability or complex classification are still challenging (ex: medical imaging)

Large Language Models

- Large Language Models caught everyone's attention (ChatGPT)
- Challenges of reducing their resources (data/power)
- May hallucinate: lack of robustness

Many other domains

Multimodal models (DALL-E, ...), Audio, Games, Video, ...

2024-04-24

Course 4: Deep Learning

└What about now?

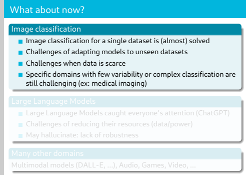

What about now?

Image classification

- Image classification for a single dataset is (almost) solved
- Challenges of adapting models to unseen datasets
- Challenges when data is scarce
- Specific domains with few variability or complex classification are still challenging (ex: medical imaging)

Large Language Models

- Large Language Models caught everyone's attention (ChatGPT)
- Challenges of reducing their resources (data/power)
- May hallucinate: lack of robustness

Many other domains

Multimodal models (DALL-E, ...), Audio, Games, Video, ...

What about now?

Image classification

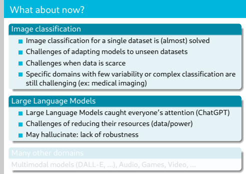
- Image classification for a single dataset is (almost) solved
- Challenges of adapting models to unseen datasets
- Challenges when data is scarce
- Specific domains with few variability or complex classification are still challenging (ex: medical imaging)

Large Language Models

- Large Language Models caught everyone’s attention (ChatGPT)
- Challenges of reducing their resources (data/power)
- May hallucinate: lack of robustness

Many other domains

Multimodal models (DALL-E, ...), Audio, Games, Video, ...



What about now?

Image classification
- Image classification for a single dataset is (almost) solved
- Challenges of adapting models to unseen datasets
- Challenges when data is scarce
- Specific domains with few variability or complex classification are still challenging (ex: medical imaging)

Large Language Models
- Large Language Models caught everyone’s attention (ChatGPT)
- Challenges of reducing their resources (data/power)
- May hallucinate: lack of robustness

Many other domains
Multimodal models (DALL-E, ...), Audio, Games, Video, ...

What about now?

Image classification

- Image classification for a single dataset is (almost) solved
- Challenges of adapting models to unseen datasets
- Challenges when data is scarce
- Specific domains with few variability or complex classification are still challenging (ex: medical imaging)

Large Language Models

- Large Language Models caught everyone’s attention (ChatGPT)
- Challenges of reducing their resources (data/power)
- May hallucinate: lack of robustness

Many other domains

Multimodal models (DALL-E, ...), Audio, Games, Video, ...

2024-04-24

Course 4: Deep Learning

└What about now?

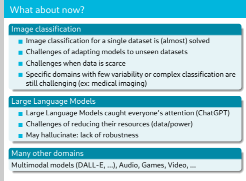

What about now?

Image classification
- Image classification for a single dataset is (almost) solved
- Challenges of adapting models to unseen datasets
- Challenges when data is scarce
- Specific domains with few variability or complex classification are still challenging (ex: medical imaging)

Large Language Models
- Large Language Models caught everyone’s attention (ChatGPT)
- Challenges of reducing their resources (data/power)
- May hallucinate: lack of robustness

Many other domains
Multimodal models (DALL-E, ...), Audio, Games, Video, ...

Focus on Large Language Models

Many models

- GPT (Open-AI)
- LLaMA (Meta)
- Gemini (Google)
- Mistral 8x7B (MistralAI)
- Many others... And more to come!

Masked Language Modeling

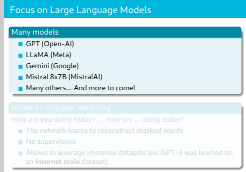
*How are **you** doing today?* → *How are ... doing today?*

- The network learns to reconstruct masked words
- No supervision!
- Allows to leverage immense datasets (ex: GPT-3 was learned on an **Internet scale** dataset)

2024-04-24

Course 4: Deep Learning

└Focus on Large Language Models


Focus on Large Language Models

Many models

- GPT (Open-AI)
- LLaMA (Meta)
- Gemini (Google)
- Mistral 8x7B (MistralAI)
- Many others... And more to come!

Masked Language Modeling

How are you doing today? → How are ... doing today?

- The network learns to reconstruct masked words
- No supervision!
- Allows to leverage immense datasets (ex: GPT-3 was learned on an Internet scale dataset)

Focus on Large Language Models

Many models

- GPT (Open-AI)
- LLaMA (Meta)
- Gemini (Google)
- Mistral 8x7B (MistralAI)
- Many others... And more to come!

Masked Language Modeling

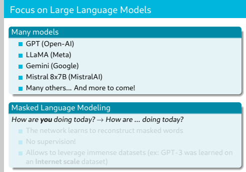
*How are **you** doing today? → How are ... doing today?*

- The network learns to reconstruct masked words
- No supervision!
- Allows to leverage immense datasets (ex: GPT-3 was learned on an **Internet scale** dataset)

2024-04-24

Course 4: Deep Learning

└Focus on Large Language Models



Focus on Large Language Models

Many models

- GPT (Open-AI)
- LLaMA (Meta)
- Gemini (Google)
- Mistral 8x7B (MistralAI)
- Many others... And more to come!

Masked Language Modeling

*How are **you** doing today? → How are ... doing today?*

- The network learns to reconstruct masked words
- No supervision!
- Allows to leverage immense datasets (ex: GPT-3 was learned on an **Internet scale** dataset)

Focus on Large Language Models

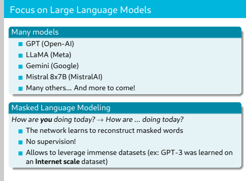
Many models

- GPT (Open-AI)
- LLaMA (Meta)
- Gemini (Google)
- Mistral 8x7B (MistralAI)
- Many others... And more to come!

Masked Language Modeling

*How are **you** doing today?* → *How are ... doing today?*

- The network learns to reconstruct masked words
- No supervision!
- Allows to leverage immense datasets (ex: GPT-3 was learned on an **Internet scale** dataset)



Focus on Large Language Models

Many models

- GPT (Open-AI)
- LLaMA (Meta)
- Gemini (Google)
- Mistral 8x7B (MistralAI)
- Many others... And more to come!

Masked Language Modeling

*How are **you** doing today?* → *How are ... doing today?*

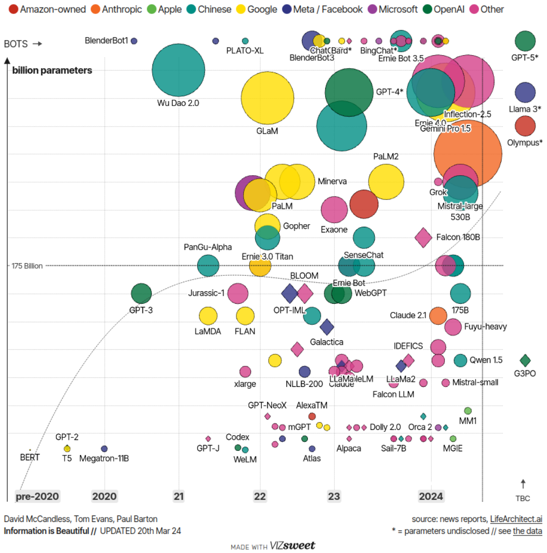
- The network learns to reconstruct masked words
- No supervision!
- Allows to leverage immense datasets (ex: GPT-3 was learned on an **Internet scale** dataset)

Large Language Models are greedy

Model sizes

- AlexNet (2012): 62 Million parameters
- GPT-3 (2020): 175 Billion parameters

Image source: `https://informationisbeautiful.net/visualizations/the-rise-of-generative-ai-large-langua`



David McCandless, Tom Evans, Paul Barton
Information is Beautiful // UPDATED 20th Mar 24

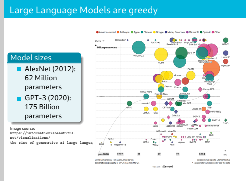
source: news reports, LifeArchitect.ai
* = parameters undisclosed // see the data

MADE WITH *VIZsweet*

2024-04-24

Course 4: Deep Learning

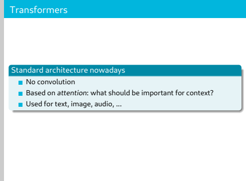
└Large Language Models are greedy


Large Language Models are greedy
Model sizes
AlexNet (2012): 62 Million parameters
GPT-3 (2020): 175 Billion parameters

Transformers

Standard architecture nowadays

- No convolution
- Based on *attention*: what should be important for context?
- Used for text, image, audio, ...



Transformers

Standard architecture nowadays

- No convolution
- Based on *attention*: what should be important for context?
- Used for text, image, audio, ...

Transformers

Standard architecture nowadays

- No convolution
- Based on *attention*: what should be important for context?
- Used for text, image, audio, ...

Transformer block

Based on 3 elements:

- Key
- Query
- Value

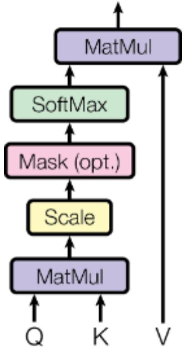
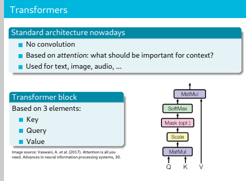


Image source: Vaswani, A. *et al.* (2017). Attention is all you need. Advances in neural information processing systems, 30.


Transformers

Standard architecture nowadays
- No convolution
- Based on *attention*: what should be important for context?
- Used for text, image, audio, ...

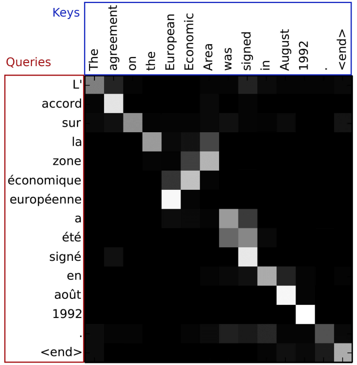
Transformer block
Based on 3 elements:
- Key
- Query
- Value

Image source: Vaswani, A. et al. (2017). Attention is all you need. Advances in neural information processing systems, 30.

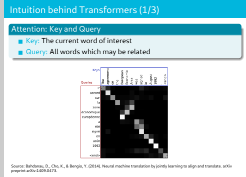
Intuition behind Transformers (1/3)

Attention: Key and Query

- Key: The current word of interest
- Query: All words which may be related



Source: Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473.

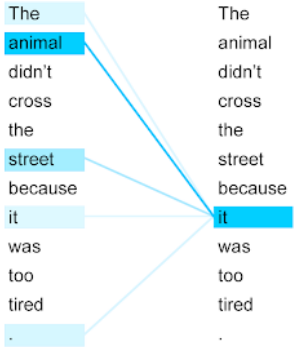

Intuition behind Transformers (1/3)
Attention: Key and Query
Key: The current word of interest
Query: All words which may be related
Source: Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473.

Attention consists in asking: which word is important to explain my current word?

Intuition behind Transformers (2/3)

From Attention to Self-Attention

In self-attention, Keys and Queries come from the same text: **context**.

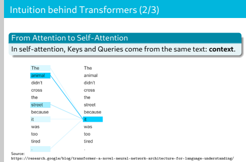


Source:
`https://research.google/blog/transformer-a-novel-neural-network-architecture-for-language-understanding/`

2024-04-24

Course 4: Deep Learning

└Intuition behind Transformers (2/3)

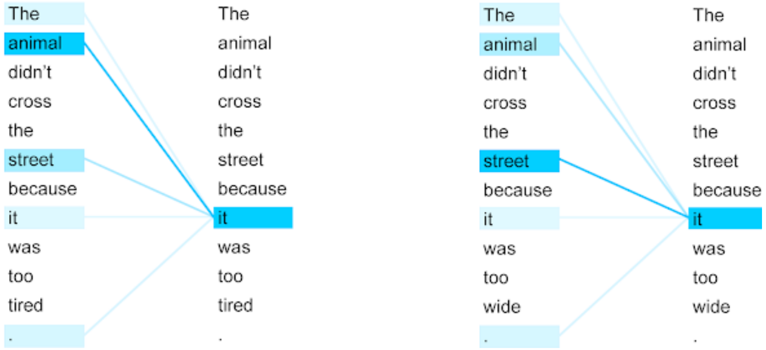


Notice how the model is able to understand the context of the “it” in the examples: “it” refers to the animal in the first example, and to the street in the second. The model has caught this behavior.

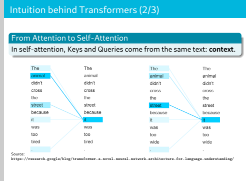
Intuition behind Transformers (2/3)

From Attention to Self-Attention

In self-attention, Keys and Queries come from the same text: **context**.



Source:
`https://research.google/blog/transformer-a-novel-neural-network-architecture-for-language-understanding/`


Intuition behind Transformers (2/3)
From Attention to Self-Attention
In self-attention, Keys and Queries come from the same text: **context**.

Notice how the model is able to understand the context of the “it” in the examples: “it” refers to the animal in the first example, and to the street in the second. The model has caught this behavior.

Intuition behind Transformers (3/3)

Transformer block

- Key and Query: Context
- Value: Modify the current work, to integrate context

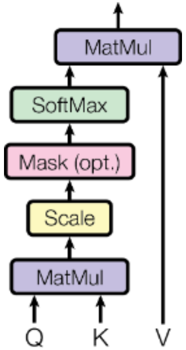
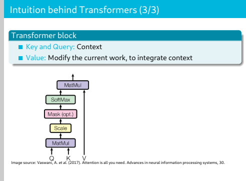


Image source: Vaswani, A. *et al.* (2017). Attention is all you need. Advances in neural information processing systems, 30.


Intuition behind Transformers (3/3)
Transformer block
Key and Query: Context
Value: Modify the current work, to integrate context
Image source: Vaswani, A. et al. (2017). Attention is all you need. Advances in neural information processing systems, 30.

Finally, a Transformer block will modify each word for it to integrate the context learned with Keys and Queries. In the end, the whole sentence is encoded through the embeddings of each word.

Intuition behind Transformers (3/3)

Transformer block

- Key and Query: Context
- Value: Modify the current work, to integrate context

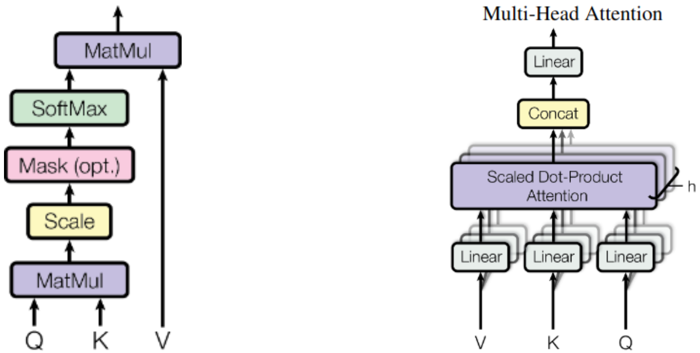
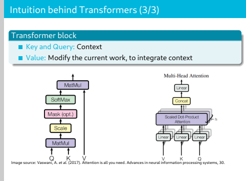


Image source: Vaswani, A. *et al.* (2017). Attention is all you need. Advances in neural information processing systems, 30.


Intuition behind Transformers (3/3)
Transformer block
Key and Query: Context
Value: Modify the current work, to integrate context
Image source: Vaswani, A. et al. (2017). Attention is all you need. Advances in neural information processing systems, 30.

Finally, a Transformer block will modify each word for it to integrate the context learned with Keys and Queries. In the end, the whole sentence is encoded through the embeddings of each word.

Repeat Transformer blocks: **Deep** model

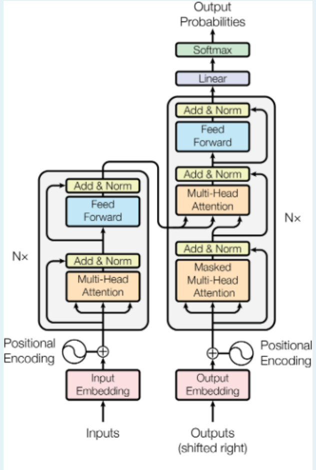
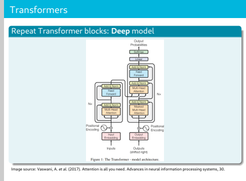


Figure 1: The Transformer - model architecture.

Image source: Vaswani, A. *et al.* (2017). Attention is all you need. Advances in neural information processing systems, 30.



doesn't need to go in details, just present the fact that blocks are repeated.

Deep Learning

Conclusion

- Deep Learning algorithms: **powerful** without feature extraction
- They require **a lot** of data to be trained
- The architecture plays an important role

Common criticisms

- Hard to interpret
- Reproduce biases from data
- May require massive amounts of energetic consumption

Going further

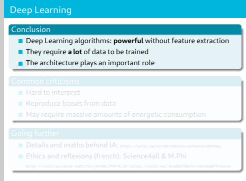
- Details and maths behind IA: https://youtu.be/aircAruvnKk?si=yOVkw0sOvDHVZbQj
- Ethics and reflexions (french): Science4all & M.Phi

  https://youtu.be/sAjm3-IaRtI?si=j41k66_FYX77L_HT | https://youtu.be/_XJsAQsTOBo?si=OFJdmqR7YvF5wJA-

2024-04-24

Course 4: Deep Learning

└Deep Learning



Deep Learning

Conclusion

- Deep Learning algorithms: **powerful** without feature extraction
- They require **a lot** of data to be trained
- The architecture plays an important role

Common criticisms

- Hard to interpret
- Reproduce biases from data
- May require massive amounts of energetic consumption

Going further

- Details and maths behind IA: https://youtu.be/aircAruvnKk?si=yOVkw0sOvDHVZbQj
- Ethics and reflexions (french): Science4all & M.Phi

  https://youtu.be/sAjm3-IaRtI?si=j41k66_FYX77L_HT | https://youtu.be/_XJsAQsTOBo?si=OFJdmqR7YvF5wJA-

Deep Learning

Conclusion

- Deep Learning algorithms: **powerful** without feature extraction
- They require **a lot** of data to be trained
- The architecture plays an important role

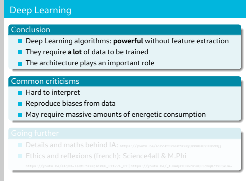
Common criticisms

- Hard to interpret
- Reproduce biases from data
- May require massive amounts of energetic consumption

Going further

- Details and maths behind IA: https://youtu.be/aircAruvnKk?si=yOVkwOsOvDHVZbQj
- Ethics and reflexions (french): Science4all & M.Phi

  https://youtu.be/sAjm3-IaRtI?si=j41k66_FYX77L_HT | https://youtu.be/_XJsAQsTOBo?si=OFJdmqR7YvF5wJA-



Deep Learning

Conclusion

- Deep Learning algorithms: **powerful** without feature extraction
- They require **a lot** of data to be trained
- The architecture plays an important role

Common criticisms

- Hard to interpret
- Reproduce biases from data
- May require massive amounts of energetic consumption

Going further

- Details and maths behind IA: https://youtu.be/aircAruvnKk?si=yOVkwOsOvDHVZbQj
- Ethics and reflexions (french): Science4all & M.Phi

  https://youtu.be/sAjm3-IaRtI?si=j41k66_FYX77L_HT | https://youtu.be/_XJsAQsTOBo?si=OFJdmqR7YvF5wJA-

Deep Learning

Conclusion

- Deep Learning algorithms: **powerful** without feature extraction
- They require **a lot** of data to be trained
- The architecture plays an important role

Common criticisms

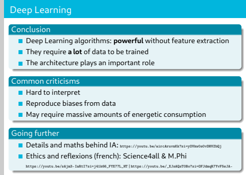
- Hard to interpret
- Reproduce biases from data
- May require massive amounts of energetic consumption

Going further

- Details and maths behind IA: `https://youtu.be/aircAruvnKk?si=yOVkw0sOvDHVZbQj`
- Ethics and reflexions (french): Science4all & M.Phi

`https://youtu.be/sAjm3-IaRtI?si=j41k66_FYX77L_HT` | `https://youtu.be/_XJsAQsTOBo?si=OFJdmqR7YvF5wJA-`

└Deep Learning



Deep Learning

Conclusion
- Deep Learning algorithms: **powerful** without feature extraction
- They require **a lot** of data to be trained
- The architecture plays an important role

Common criticisms
- Hard to interpret
- Reproduce biases from data
- May require massive amounts of energetic consumption

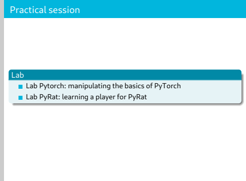
Going further
- Details and maths behind IA: https://youtu.be/aircAruvnKk?si=yOVkw0sOvDHVZbQj
- Ethics and reflexions (french): Science4all & M.Phi

https://youtu.be/sAjm3-IaRtI?si=j41k66_FYX77L_HT | https://youtu.be/_XJsAQsTOBo?si=OFJdmqR7YvF5wJA-

Practical session

Lab

- Lab Pytorch: manipulating the basics of PyTorch
- Lab PyRat: learning a player for PyRat



Practical session

Lab

- Lab Pytorch: manipulating the basics of PyTorch
- Lab PyRat: learning a player for PyRat