Course 4: Deep Learning



Cour

Course 4: Deep Learning

Course 4: Deep Learning



Summary

Last session

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

Today's session

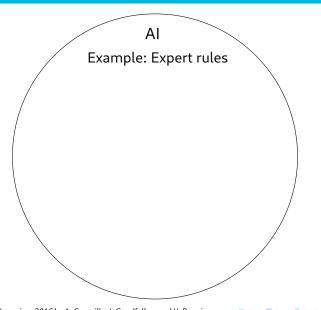
- Multi-Layer Perceptron
- Convolutional Neural Networks
- Transformers

Course 4: Deep Learning

Last session

Unsupervised learningunsubsed data are from
unsubsed data are from
unsubsed data are
Unsupervised learningunsubsed data are from
unsubsed data are
Unsupervised learningunsubsed data are from
unsubsed data are
Unsupervised learningunsubsed data are
Unsupervised learningunsubsed data are
Unsupervised learningunsupervised learningunsuperv

Global overview...



Course 4: Deep Learning

-Global overview...

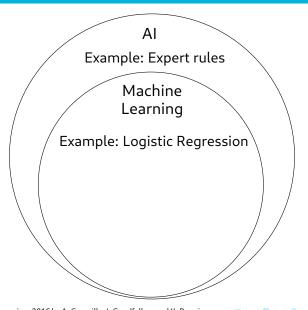


Deep Learning is a particular case of Machine Learning.

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



Global overview...



Course 4: Deep Learning

-Global overview...

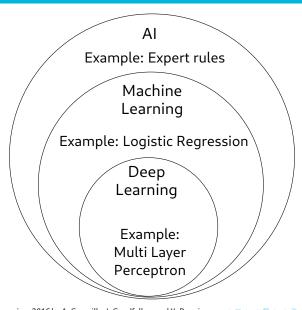


Deep Learning is a particular case of Machine Learning.

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

Course 4: Deep Learning

Global overview...



Course 4: Deep Learning

-Global overview...



Deep Learning is a particular case of Machine Learning.

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

Course 4: Deep Learning

Deep Learning in a nutshell (1/3)

Course 4: Deep Learning

-Deep Learning in a nutshell (1/3)

Deep Learning in a nutshell (1/3) Optimize Optimize State of the American State of the State of State

Definition

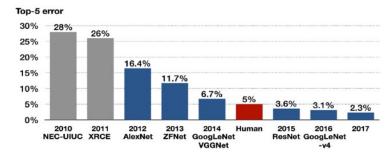
Using deep Artificial Neural Networks.

We generally talk about "Neural Networks instead of "Artificial Neural Networks" (but this is the most correct terminology!)

The strength of Deep Learning lies in using a lot (a lot) of data.

Deep Learning in a nutshell (2/3)

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



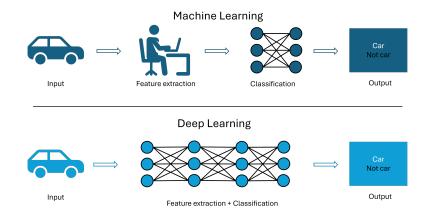
Course 4: Deep Learning



—Deep Learning in a nutshell (2/3)

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 (3/3)



 $In spired from \ https://www.softwaretestinghelp.com/data-mining-vs-machine-learning-vs-ai/spired from \ https://www.softwaretestinghelp.com/data-mining-vs-machine-lear$



Course 4: Deep Learning

Deep Learning in a nutshell (3/3)



Outline

2024-

Course 4: Deep Learning
—Multi-layer Perceptron -Multi-layer Perceptron

-Outline

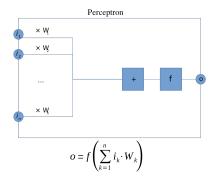
1 Multi-layer Perceptron

Perceptron

IMT-Atlantique

1943, implementation in 1957

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

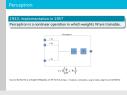


Source: By Mat the w at English Wikipedia, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=23766733

Course 4: Deep Learning



Course 4: Deep Learning
—Multi-layer Perceptron
—Perceptron



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.

Perceptron

1943, implementation in 1957

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

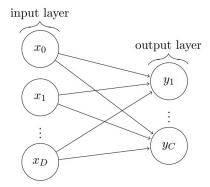
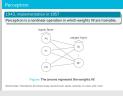


Figure: The arrows represent the weights W.

Course 4: Deep Learning Multi-layer Perceptron 2024 -Perceptron

0



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.

Source: https://davidstutz.de/illustrating-convolutional-neural-networks-in-latex-with-tikz/ = | IMT-Atlantique Course 4: Deep Learning

Optimizing the weights

Loss

■ Prediction:
$$y = f\left(\sum_{d=0}^{D} x_d W_d\right)$$

- Ground truth: ŷ
- 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: $W \leftarrow W \eta \frac{\partial J(W)}{\partial W}$

Course 4: Deep Learning
—Multi-layer Perceptron
—Optimizing the weights



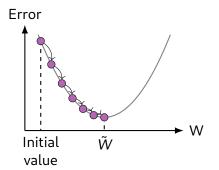
The Perceptron is a matrix multiplication, followed by a nonlinear function f(). It is important to say that the Perceptron is old!! Hence, AI is not a brand new thing, but an old research domain.

Gradient descent

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
- \blacksquare η gives the size of the step



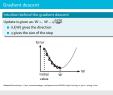
Adapted from https://tex.stackexchange.com/questions/561921/replicating-a-plot-using-tikz

Z-10-470: — Mu

Course 4: Deep Learning

Multi-layer Perceptron

Gradient descent



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.

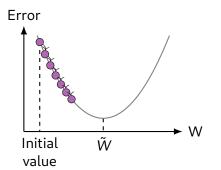
Gradient descent

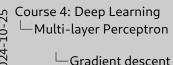
Intuition behind the gradient descent

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

- lacksquare $\partial J(W)$ gives the direction
- lacksquare 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.

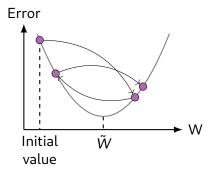
Gradient descent

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
- lacksquare η gives the size of the step

Large step:



Course 4: Deep Learning

Multi-layer Perceptron

Gradient descent

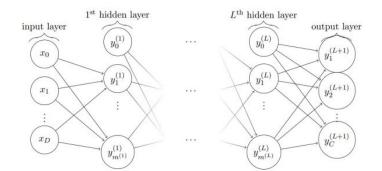


The gradient follows the increase of the error function, hence the inverse of the gradient follows its decrease. Parameter η , the learning rate, gives the size of the step to take at each iteration. If too small, the model will slowly converge. If too large, the model can be unstable and never reach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

Multi-Layer Perceptron

Multi-Layer Perceptron (= fully-connected network)

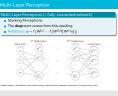
- Stacking Perceptrons.
- The *deep* term comes form this stacking
- Prediction: $y = f(W^{(L)} \cdots f(W^{(2)} f(W^{(1)} x)))$



Course 4: Deep Learning

Multi-layer Perceptron

Multi-Layer Perceptron



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

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

Definitions

$$\mathbf{v}^{(l+1)} = f(\mathbf{W}^{(l)}\mathbf{v}^{(l)} + \mathbf{b}^{(l)}) = \phi^{(l)}(\mathbf{v}^{(l)})$$

- **Each** building block $\mathbf{y}^{(l+1)}$ is called a **layer**.

Course 4: Deep Learning Multi-layer Perceptron -Neural Networks

2024

 $\mathbf{v}^{(l+1)} = f(\mathbf{W}^{(l)}\mathbf{v}^{(l)} + \mathbf{b}^{(l)}) = \phi^{(l)}(\mathbf{v}^{(l)})$

■ Each building block v(1+1) is called a laver

Definitions

$$\mathbf{v}^{(l+1)} = f(\mathbf{W}^{(l)}\mathbf{v}^{(l)} + \mathbf{b}^{(l)}) = \phi^{(l)}(\mathbf{v}^{(l)})$$

- **Each** building block $\mathbf{y}^{(l+1)}$ is called a **layer**.
- One element i of a layer $(y_i^{(l)})$ is called a **neuron** (both as input and output).
- The nonlinear function *f* is called the **activation function**.
- W^(l) are called weights.
- **b**(l) is called the **bias**.

Note: while each layer can have a different activation function f, it is standard that each layer uses the same.

Course 4: Deep Learning

Multi-layer Perceptron

Neural Networks

Neural Networks $y^{n+1} = (y^{n}(y^n + y^n)^n) - \phi^n(y^n)$ $= 5 \text{th foliation block } y^{n+1} \text{is called a layer.}$ $= 5 \text{th foliation block } y^{n+1} \text{is called a layer.}$ $= 5 \text{th foliation of of layer } (y^n)^n \text{ is called a layer.}$ $= 5 \text{th foliation of a layer } (y^n)^n \text{ is called a layer.}$ $= 5 \text{th foliation of a layer } (y^n)^n \text{ is called a layer.}$ $= 5 \text{th foliation of a layer } (y^n)^n \text{ is called a layer.}$ = 5 th foliation of a layer. = 5 th foliation of a layer. = 5 th foliation of a layer.

Definitions

$$\mathbf{v}^{(l+1)} = f(\mathbf{W}^{(l)}\mathbf{v}^{(l)} + \mathbf{b}^{(l)}) = \phi^{(l)}(\mathbf{v}^{(l)})$$

- **Each** building block $\mathbf{y}^{(l+1)}$ is called a **layer**.
- One element i of a layer $(y_i^{(l)})$ is called a **neuron** (both as input and output).
- The nonlinear function f is called the activation function.

Course 4: Deep Learning Multi-layer Perceptron -Neural Networks $\mathbf{v}^{(i+1)} = f(\mathbf{W}^{(i)}\mathbf{v}^{(i)} + \mathbf{b}^{(i)}) = \phi^{(i)}(\mathbf{v}^{(i)})$

Each building block v⁽¹⁺¹⁾ is called a laver

The nonlinear function f is called the activation function

IMT-Atlantique Course 4: Deep Learning

Definitions

$$\mathbf{v}^{(l+1)} = f(\mathbf{W}^{(l)}\mathbf{v}^{(l)} + \mathbf{b}^{(l)}) = \phi^{(l)}(\mathbf{v}^{(l)})$$

- **Each** building block $\mathbf{y}^{(l+1)}$ is called a **layer**.
- One element i of a layer $(y_i^{(l)})$ is called a **neuron** (both as input and output).
- The nonlinear function *f* is called the **activation function**.
- **W**^(l) are called **weights**.
- **b**(l) is called the bias.

Note: while each layer can have a different activation function f, it is standard that each layer uses the same.

Course 4: Deep Learning

Multi-layer Perceptron

Neural Networks

Neural Networks

Definitions $\mathbf{y}^{(i+1)} = f(\mathbf{W}^{(i)}\mathbf{y}^{(i)} + \mathbf{b}^{(i)}) = \phi^{(i)}(\mathbf{y}^{(i)})$ If Each building block $\mathbf{y}^{(i)}$ is called a layer.

If One element is $\mathbf{y}^{(i)}$ is called a nerver (both as output).

If $\mathbf{y}^{(i)}$ is a supervised on the property of th

Definitions

$$\mathbf{v}^{(l+1)} = f(\mathbf{W}^{(l)}\mathbf{v}^{(l)} + \mathbf{b}^{(l)}) = \phi^{(l)}(\mathbf{v}^{(l)})$$

- **Each** building block $y^{(l+1)}$ is called a **layer**.
- One element i of a layer $(y_i^{(l)})$ is called a **neuron** (both as input and output).
- The nonlinear function *f* is called the **activation function**.
- **W**^(l) are called **weights**.
- **b**(l) is called the **bias**.

Note: while each layer can have a different activation function f, it is standard that each layer uses the same.



Course 4: Deep Learning

Multi-layer Perceptron

Neural Networks

Neural Networks $Y^{(i,j)} = \{(X^{(i)}y^{(i)} + b^{(i)}) - \phi^{(i)}y^{(i)}\}$ is Each building block $y^{(i+1)}$ is called a lawy. If not obtain y is the scale of a lawy of $y^{(i)}$ is called a neuron Both as input and part y. If the called the activation function y is the nonlinear fraction if it is called the activation function.

Definitions

$$\mathbf{v}^{(l+1)} = f(\mathbf{W}^{(l)}\mathbf{v}^{(l)} + \mathbf{b}^{(l)}) = \phi^{(l)}(\mathbf{v}^{(l)})$$

- **Each** building block $\mathbf{y}^{(l+1)}$ is called a **layer**.
- One element i of a layer $(y_i^{(l)})$ is called a **neuron** (both as input and output).
- The nonlinear function *f* is called the **activation function**.
- **W**^(l) are called **weights**.
- **b**(l) is called the **bias**.

Note: while each layer can have a different activation function f, it is standard that each layer uses the same.

Course 4: Deep Learning

Multi-layer Perceptron

-Neural Networks

Neural Networks Definitions $y^{(i+1)} = (y_i w^{(i)} y^{(i)} + b^{(i)}) = \phi^{(i)}(y^{(i)})$ If Each building block $y^{(i+1)}$ is called a layer. If Does element if a layer $(y^{(i)})$ is called a newern (both as in The nonlinear function if is called the activation function $y^{(i)}$ is called the activation function.

N/I is called the blas

Note: while each layer can have a different activation function f, it is standard that each layer uses the same.

Why is it called Neural Network?

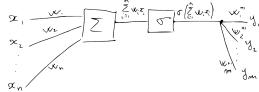
"Neurons" may be seen as **loosely** inspired from the human brain.



Wikipedia,
https://en.wikipedia.org/wiki/Neural_

network_(machine_learning)#/media/

File:Neuron3.png



Detail of one neuron in an Artificial Neural Network.

* Late note: I forgot the bias in the graphic.

This is an **analogy**, artificial neural networks are **not** following the human brain (in general).



Course 4: Deep Learning

Multi-layer Perceptron

10-

─Why is it called Neural Network?

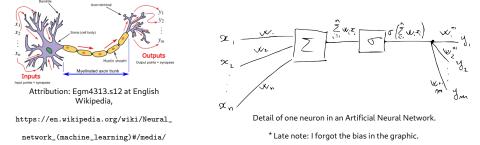


IMT-Atlantique Course 4: Deep Learning 12/31

Why is it called Neural Network?

File:Neuron3.png

"Neurons" may be seen as **loosely** inspired from the human brain.



This is an **analogy**, artificial neural networks are **not** following the human brain (in general).

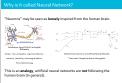


Course 4: Deep Learning

Multi-layer Perceptron

0

—Why is it called Neural Network?

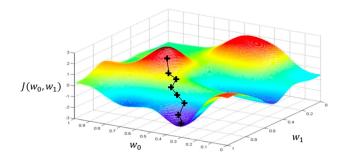


IMT-Atlantique Course 4: Deep Learning 12/31

Gradient descent for deep neural networks

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)



Course 4: Deep Learning

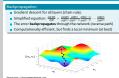
Source: http://introtodeeplearning.com/

IMT-Atlantique

Course 4: Deep Learning

Multi-layer Perceptron

-Gradient descent for deep neural networks



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

Batch

Course 4: Deep Learning

Multi-layer Perceptron

Batch

a. The reamples are divided in histories (small except): a. The reamples are divided in histories (small except): a. Allows one of a rain without loading the whole dataset in memory. B. Accelerate the learning phase

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

IMT-Atlantique

Course 4: Deep Learning

14/31

Limits of MLP

Course 4: Deep Learning

Multi-layer Perceptron

Limits of MLP

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

Limits of MLP

Limits of Multi-Layer Receptors

Computationally heavy for large layers

Computationally heavy for large layers

Layer eliaber of presenter process or overfitting

No notion of another is the layer, excepting is a vector

Outline

Multi-laver Perceptror

2 Convolutional Neural Networks (CNN)

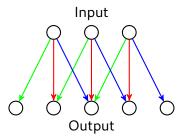
3 Transformers



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



IMT-Atlantique



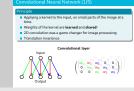
ロト 4 何 ト 4 き ト 4 き ト き めのぐ

Course 4: Deep Learning 16

Course 4: Deep Learning

Convolutional Neural Networks (CNN)

-Convolutional Neural Network (1/5)



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.

Example of 2D convolution:

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

Course 4: Deep Learning
Convolutional Neural Networks (CNN)
Convolutional Neural Network (2/5)

Example of 2D convolution:

$$\begin{pmatrix}
0 & 1 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 1 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 0 & 0 & 0
\end{pmatrix}$$

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

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

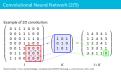
 $Source: \verb|https://tex.stackexchange.com/questions/437007/drawing-a-convolution-with-tikz| \\$

Course 4: Deep Learning
Convolutional Neural Networks (CNN)
Convolutional Neural Network (2/5)

Example of 2D convolution:

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

Course 4: Deep Learning
Convolutional Neural Networks (CNN)
Convolutional Neural Network (2/5)



Example of 2D pooling:

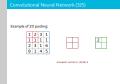
1	2	3	1
1	1	1	1
2	3	1	6
8	1	4	5





maxpool, kernel 2, stride 2

Course 4: Deep Learning
Convolutional Neural Networks (CNN)
Convolutional Neural Network (3/5)



10-

Example of 2D pooling:

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





maxpool, kernel 2, stride 2

Course 4: Deep Learning
Convolutional Neural Networks (CNN)
Convolutional Neural Network (3/5)



10-

Example of 2D pooling:

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





maxpool, kernel 2, stride 2

Course 4: Deep Learning
Convolutional Neural Networks (CNN)
Convolutional Neural Network (3/5)



10-

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

Course 4: Deep Learning
Convolutional Neural Networks (CNN)
Convolutional Neural Network (3/5)



10-

Convolutional Neural Network (4/5)

And repeat...

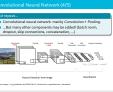
- Convolutional neural network: mainly Convolution + Pooling.
- ...But many other components may be added! (batch norm, dropout, skip connections, concatenation, ...)



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

IMT-Atlantique Course 4: Deep Learning 19/31

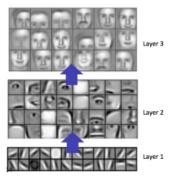
Course 4: Deep Learning
Convolutional Neural Networks (CNN)
Convolutional Neural Network (4/5)



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



Course 4: Deep Learning

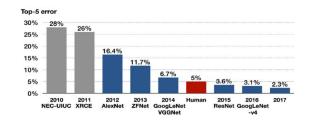
Convolutional Neural Networks (CNN)

-Convolutional Neural Network (5/5)



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...

IMT-Atlantique

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

Course 4: Deep Learning

■ The use of GPUs for computation

Course 4: Deep Learning Convolutional Neural Networks (CNN)

> -What happened in 2012? training, very standard in Deep Learning

- Clever tricks (ex: data augmentation, i.e. altering image during
- The use of GPUs for computation

Outline

Course 4: Deep Learning
Transformers 2024-

-Outline

■ Transformers

3 Transformers

State-of-the-Art nowadays

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, ...

Course 4: Deep Learning

Transformers

-State-of-the-Art nowadays

State-of-the-An nowadays

Image dasolication

I image dasolication for a single dataset it planned; solved

I challenges of adapting models to unseen datasets

I challenges when data is scarce

I poper datasets

I challenges the model analysing

I challenges the model analy

IMT-Atlantique

Course 4: Deep Learning

22/31

State-of-the-Art nowadays

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)

Course 4: Deep Learning

- Challenges of reducing their resources (data/power)
- May hallucinate: lack of robustness

IMT-Atlantique

Course 4: Deep Learning Transformers -State-of-the-Art nowadays Image classification for a single dataset is (almost) solve

 Challenges of adapting models to unseen datasets Challenges when data is scarce

Specific domains with few variability or complex classification are till challenging (ex: medical imaging)

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

State-of-the-Art nowadays

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)

Course 4: Deep Learning

- Challenges of reducing their resources (data/power)
- May hallucinate: lack of robustness

Many other domains

IMT-Atlantique

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

Course 4: Deep Learning

Transformers

-State-of-the-Art nowadays

e-of-the-Art nowad

ge classification

Challenges of adapting models to unseen datasets Challenges when data is scarce

 Challenges when data is scarce
 Specific domains with few variability or complex classification are still challenging (ex: medical imaging)

arge Language Models

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

may natiocinate: lack of robustness

nodal 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? \rightarrow 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)

Course 4: Deep Learning

Transformers

-Focus on Large Language Models

cus on Large Language Models

GPT (Open-Al)

LLaMA (Meta)
Gemini (Google)

Mistral 8x7B (MistralAl)
 Many others... And more to come!

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

■ The network learns to reconstruct masked words

The network learns to reconstruct masked words
 No supposition!

Allows to leverage immense datasets (ex: GPT-3 w

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? \rightarrow 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)

4□ × 4♬ × 4 분 × 4 분 × 9 Q (~

Course 4: Deep Learning

Transformers

-Focus on Large Language Models

models
PT (Open-Al)
LaMA (Meta)

LLaMA (Meta)
Gemini (Google)
Mistral 8x7B (MistralAl)

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

No supervision!
Allows to leverage immense datasets (ex: G

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? \rightarrow 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)

+ 다 + 4 라 + 4 분 + 1 분 보 의 수 다 + 1 년 + 1 분 보 가 수 다 + 1 년 + 1 분 보 가 수 다 + 1 년 + 1 분 보 가 수 다 + 1 분 보 가 수 다 + 1 분 보 가 보 보 가 보 <ul

Course 4: Deep Learning

Transformers

-Focus on Large Language Models

us on Large Language Models

- GPT (Open-AI)
 LLaMA (Meta)
- LLaMA (Meta)
 Gemini (Google)
- Mistral 8x7B (MistralAI)
- How are you doing today? → How are ... doing today?
- The network learns to reconstruct masked we
- No supervision!
- Allows to leverage immense datasets (ex: GPT-3 was learned or
- an Internet scale dataset)

Large Language Models are greedy

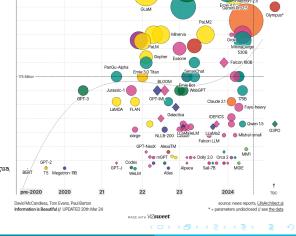
billion parameters

Model sizes

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

lmage source: https://informationisbeautiful.

net/visualizations/ the-rise-of-generative-ai-large-langua



■ Amazon-owned Anthropic Apple Chinese Google Meta / Facebook Microsoft OpenAl Other

Wu Dao 2.0

Course 4: Deep Learning

Transformers

Large Language Models are greedy



Llama 3*

0

Transformers

Course 4: Deep Learning

Transformers

Transformers

Gardard architecture nowadays In No convolution In Board on attentions what should be important for context? I lead for fact, image, andfo, ...

Standard architecture nowadays

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

Transformers

Standard architecture nowadays

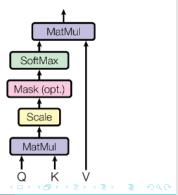
- No convolution
- Based on attention: what should be important for context?
- Used for text, image, audio, ...

Transformer block

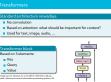
Based on 3 elements:

- Key
- Query
- Value

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



Course 4: Deep Learning
Transformers
Transformers

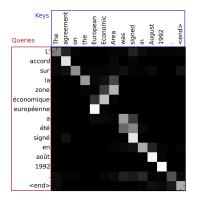


Intuition behind Transformers (1/3)

Attention: Key and Query

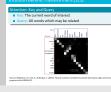
IMT-Atlantique

- 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. Course 4: Deep Learning

Course 4: Deep Learning -Transformers



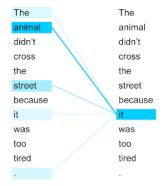
-Intuition behind Transformers (1/3)

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/



Course 4: Deep Learning

Transformers

Intuition behind Transformers (2/3)

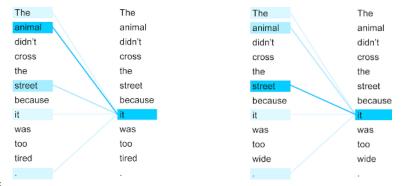


Notice how the model is able to understand the context of the "it" in the examples: "it" refers to the animal in the first example, and to the street in the second. The model has caught this behavior.

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/



Course 4: Deep Learning

Transformers



└─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.

IMT-Atlantique

Course 4: Deep Learning

27/31

Intuition behind Transformers (3/3)

Transformer block

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

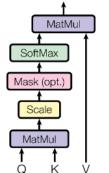


Image source: Vaswani, A. et al. (2017). Attention is all you need. Advances in neural information processing systems, 30.



Course 4: Deep Learning

Transformers

Transformer Book

If to you Connect

If to you Conn

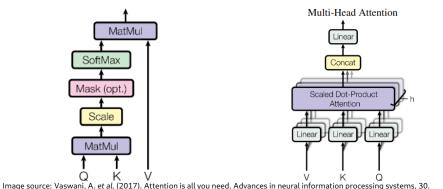
—Intuition behind Transformers (3/3)

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)

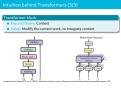
Transformer block

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



Course 4: Deep Learning
Transformers

-Intuition behind Transformers (3/3)



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.

Transformers

Repeat Transformer blocks: **Deep** model

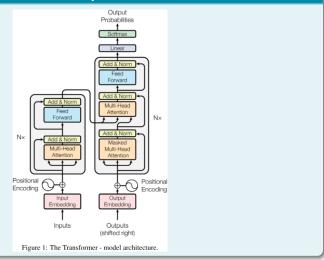


Image source: Vaswani, A. et al. (2017). Attention is all you need. Advances in neural information processing systems, 30.

Course 4: Deep Learning

Transformers

Transformers



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

IMT-Atlantique Course 4: Deep Learning 29 / 31

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
- Ethics and reflexions (french): Science4all & M.Phi

https://woutu.be/HbFadtOys4k|https://woutu.be/ XJsAOsTOBo

Course 4: Deep Learning

Transformers

Deep Learning

Conclusion

I Deep Learning algorithms: powerful without fou

I Deep Learning algorithms: powerful without fou

IT They require a let of data to be trained

If The architecture plays an important role

Conclusion of the conclusi

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
- Ethics and reflexions (french): Science4all & M.Phi

https://voutu.be/HbFadtOys4k|https://voutu.be/ XJsAOsTOBo

Course 4: Deep Learning

Transformers

Deep Learning

eep Learning
anclusion
Deep Learning algorithms: powerful without
They require a lot of data to be trained
The architecture plays an important role

May require massive amounts of energetic consumpting further

Renrodure biases from data

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
- Ethics and reflexions (french): Science4all & M.Phi

https://youtu.be/HbFadt0xs4k|https://youtu.be/_XJsAQsT0Bo

Course 4: Deep Learning
Transformers
Deep Learning

eep Learning

microscomerous depositions personnel without feature extraction

to Deep Learning algorithms personnel without feature extraction

to The architecture plays an important role

common tracticums

a The architecture plays an important role

common tracticums

a "Beyonduce Saless from data

a May require muscles amounts of energetic consumption

■ Ethics and reflexions (french): Science4all & M.Phi

Practical session

Course 4: Deep Learning

Transformers

Practical session

Lab

II Lab Pytorch: manipulating the basics of PyTorch

II Lab Modality: try a first baseline on your modality

Lab

- Lab Pytorch: manipulating the basics of PyTorch
- Lab Modality: try a first baseline on your modality