

# Introduction to course "Efficient Deep Learning"



**IMT Atlantique**  
Bretagne-Pays de la Loire  
École Mines-Télécom

February 2nd 2022

# What is AI?

## AI

- **Intelligence:** ability to **extract knowledge** from observations
- This knowledge is used to **solve tasks in different contexts and environments**

### Memorizing (explicit)

- Memorize algorithms
- 20th century preferred methodology
- **Pros:** explicit control
- **Cons:** requires explicit solutions

Not AI

### Generalization (implicit)

- Infer process from observations
- Guessing game
- **Pros:** universally applicable
- **Cons:** found solution might not be right

AI

# What is AI?

## AI

- **Intelligence:** ability to **extract knowledge** from observations
- This knowledge is used to **solve tasks in different contexts and environments**

### Memorizing (explicit)

- Memorize algorithms
- 20th century preferred methodology
- **Pros:** explicit control
- **Cons:** requires explicit solutions

Not AI

### Generalization (implicit)

- Infer process from observations
- Guessing game
- **Pros:** universally applicable
- **Cons:** found solution might not be right

AI

# What is AI?

## AI

- **Intelligence:** ability to **extract knowledge** from observations
- This knowledge is used to **solve tasks in different contexts and environments**

## Memorizing (explicit)

- Memorize algorithms
- 20th century preferred methodology
- **Pros:** explicit control
- **Cons:** requires explicit solutions

**Not AI**

## Generalization (implicit)

- Infer process from observations
- Guessing game
- **Pros:** universally applicable
- **Cons:** found solution might not be right

**AI**

# Machine learning and deep learning

## Machine learning

- **Supervised:** Infer a function from inputs/outputs

## Difficulties

- Ill-posed problem (infinity of potential solutions)
- **Main approach:** seek for particular solutions

## Deep Learning

- Express solutions as assembly of atomic functions called layers
  - **Compositional approach**
- Tune all atomic functions altogether
  - **End-to-end learning**
- Optimize using stochastic gradient descent variants
  - **Differentiable algorithmic**

**Ambition: become the new informatics**

# Machine learning and deep learning

## Machine learning

- **Supervised:** Infer a function from inputs/outputs

## Difficulties

- Ill-posed problem (infinity of potential solutions)
- **Main approach:** seek for particular solutions

## Deep Learning

- Express solutions as assembly of atomic functions called layers
  - **Compositional approach**
- Tune all atomic functions altogether
  - **End-to-end learning**
- Optimize using stochastic gradient descent variants
  - **Differentiable algorithmic**

Ambition: become the new informatics

# Machine learning and deep learning

## Machine learning

- **Supervised:** Infer a function from inputs/outputs

## Difficulties

- Ill-posed problem (infinity of potential solutions)
- **Main approach:** seek for particular solutions

## Deep Learning

- Express solutions as assembly of atomic functions called layers
  - **Compositional approach**
- Tune all atomic functions altogether
  - **End-to-end learning**
- Optimize using stochastic gradient descent variants
  - **Differentiable algorithmic**

# Machine learning and deep learning

## Machine learning

- **Supervised:** Infer a function from inputs/outputs

## Difficulties

- Ill-posed problem (infinity of potential solutions)
- **Main approach:** seek for particular solutions

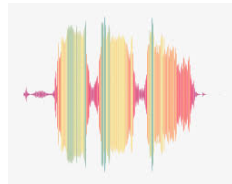
## Deep Learning

- Express solutions as assembly of atomic functions called layers
  - **Compositional approach**
- Tune all atomic functions altogether
  - **End-to-end learning**
- Optimize using stochastic gradient descent variants
  - **Differentiable algorithmic**

**Ambition: become the new informatics**



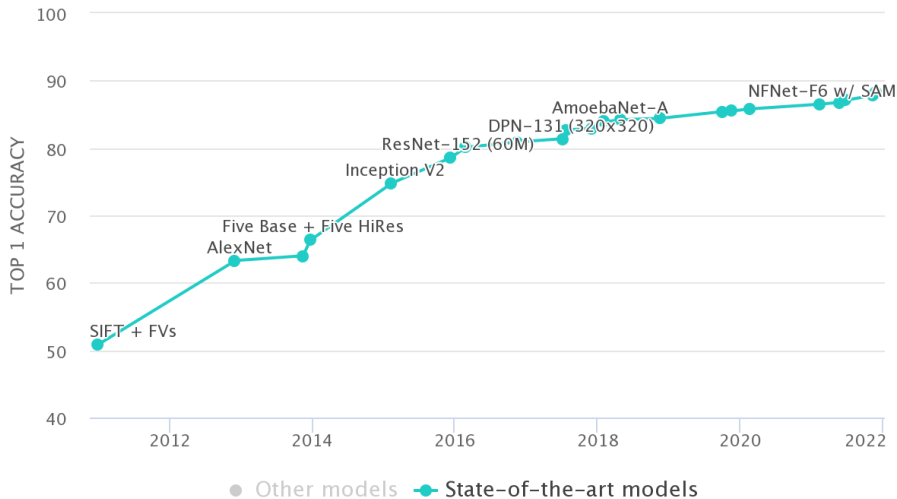
# Main results



## Your AI pair programmer

With GitHub Copilot, get suggestions for whole lines or entire functions right inside your editor.

# Example : Image Classification



source : <https://paperswithcode.com/sota/image-classification-on-imagenet>

# Limitation : computations

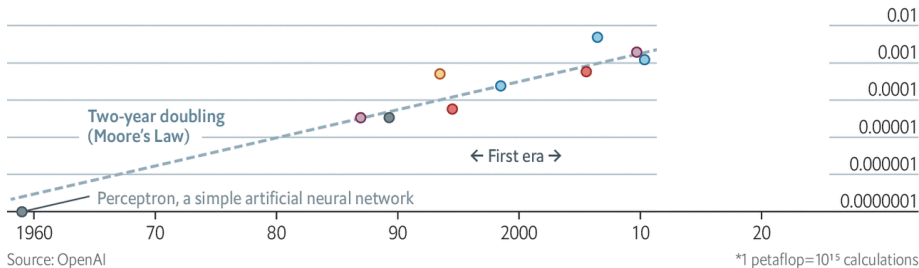
## Deep and steep

Computing power used in training AI systems

Days spent calculating at one petaflop per second\*, log scale

By fundamentals

- Language
- Speech
- Vision
- Games
- Other



# Limitation : computations

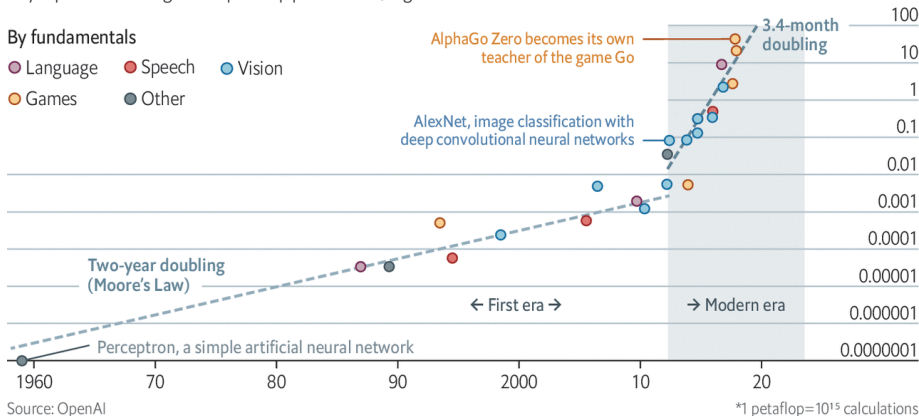
## Deep and steep

Computing power used in training AI systems

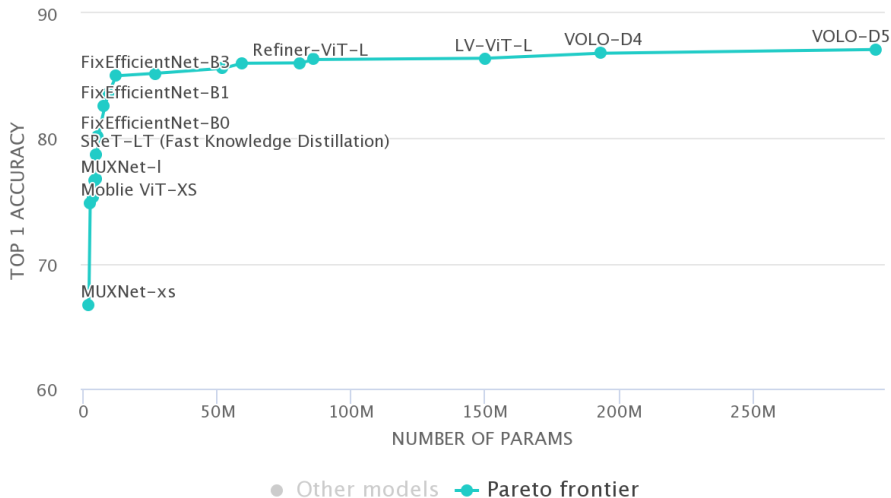
Days spent calculating at one petaflop per second\*, log scale

By fundamentals

- Language
- Speech
- Vision
- Games
- Other



# Number of parameters of Image Classification DL



source : <https://paperswithcode.com/sota/image-classification-on-imagenet>

# Making deep learning more efficient

## Why ?

- AI applications on Embedded system / Edge devices
- "Low-tech" AI with limited ressources, no cloud computing

## Problems

- Power consumption of training and inference
- Memory requirements
- Computational power requirements
- Latency

## How ?

- Reduce the number of overall parameters
- Reduce the number of computations needed
- Research on more efficient learning mechanisms

# Making deep learning more efficient

## Why ?

- AI applications on Embedded system / Edge devices
- "Low-tech" AI with limited ressources, no cloud computing

## Problems

- Power consumption of training and inference
- Memory requirements
- Computational power requirements
- Latency

## How ?

- Reduce the number of overall parameters
- Reduce the number of computations needed
- Research on more efficient learning mechanisms

# Making deep learning more efficient

## Why ?

- AI applications on Embedded system / Edge devices
- "Low-tech" AI with limited ressources, no cloud computing

## Problems

- Power consumption of training and inference
- Memory requirements
- Computational power requirements
- Latency

## How ?

- Reduce the number of overall parameters
- Reduce the number of computations needed
- Research on more efficient learning mechanisms



## Examples of challenges

- Micronet at NeurIPS 2019
- Low Power Computer Vision (since 2015)
- DCASE Task 1 challenges 2020 and 2021

## MicroNet Challenge

Hosted at NeurIPS 2019

Leaderboard

Overview

Scoring & Submission

### Announcements

1. Join the MicroNet Challenge Google Group to chat with other competitors ([link](#))

### Overview

Contestants will compete to build the most efficient model that solves the target task to the specified quality level. The competition is focused on efficient inference, and uses a theoretical metric rather than measured inference speed to score entries. We hope that this encourages a mix of submissions that are useful on today's hardware and that will also guide the direction of new hardware development.

# Efficient Deep Learning Challenges

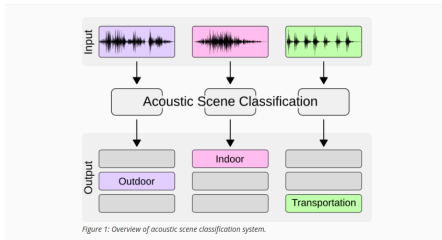
## Examples of challenges

- Micronet at NeurIPS 2019
- Low Power Computer Vision (since 2015)
- DCASE Task 1 challenges 2020 and 2021



### Low-Complexity Acoustic Scene Classification Subtask B

This subtask is concerned with the classification of audio into three major classes: indoor, outdoor, and transportation. The task targets **low complexity** solutions for the classification problem in terms of model size and uses audio recorded with a single device (device A).



source : [dcase.community](https://dcase.community)

# Efficient Deep Learning Challenges

## Examples of challenges

- Micronet at NeurIPS 2019
- Low Power Computer Vision (since 2015)
- DCASE Task 1 challenges 2020 and 2021

Rank	Submission information		Evaluation dataset			Acoustic model				System	
	Submission label	Technical Report	Official system rank	Accuracy	Logloss	Parameters	Non-zero parameters	Sparsity	Size (KB)	Complexity management	
1	Koutini_CPJKU_task1b_2		1	96.5 %	0.101	345k	247k	0.284	483.5	pruning float16	
2	Koutini_CPJKU_task1b_4		2	96.2 %	0.105	556k	249k	0.552	487.1	float16 smaller width/depth	
3	Hu_GT_task1b_3		3	96.0 %	0.122	122k	122k	0	490.0	int8 quantization	
4	McDonnell_USA_task1b_3		4	95.9 %	0.117	3M	3M	0	486.7	1-bit quantization	
5	Hu_GT_task1b_1		7	95.8 %	0.357	94k	94k	0	375.0	int8 quantization	
5	Hu_GT_task1b_4		5	95.8 %	0.131	125k	125k	0	499.0	int8 quantization	
5	McDonnell_USA_task1b_4		6	95.8 %	0.119	3M	3M	0	486.7	1-bit quantization	
6	Koutini_CPJKU_task1b_3		8	95.7 %	0.113	242k	242k	0	473.8	float16 smaller width/depth	
7	Hu_GT_task1b_2		10	95.5 %	0.367	122k	122k	0	490.0	int8 quantization	
7	McDonnell_USA_task1b_2		9	95.5 %	0.118	3M	3M	0	486.7	1-bit quantization	

source : [dcase.community](https://dcase.community)

# Course organisation

## Sessions

- 1 Intro Deep Learning,
- 2 Data Augmentation and Self Supervised Learning,
- 3 Quantization,
- 4 Pruning,
- 5 Factorization,
- 6 Distillation,
- 7 Embedded SW / HW for DL.
- 8 Presentations for challenge.

## Lab Sessions and Challenge

By groups of two, you are given a machine with complete access.

## Sessions schedule

Each session has (roughly) the same structure:

- **Short written eval** about the previous lesson (10 min),
- Short lesson (20 to 40 min),
- Lab Session,
- Project,
- Sessions 3, 5 and final include **students' presentations**.