

Introduction to AI

Stéphane Canu

https://github.com/StephaneCanu/Introduction_to_AI

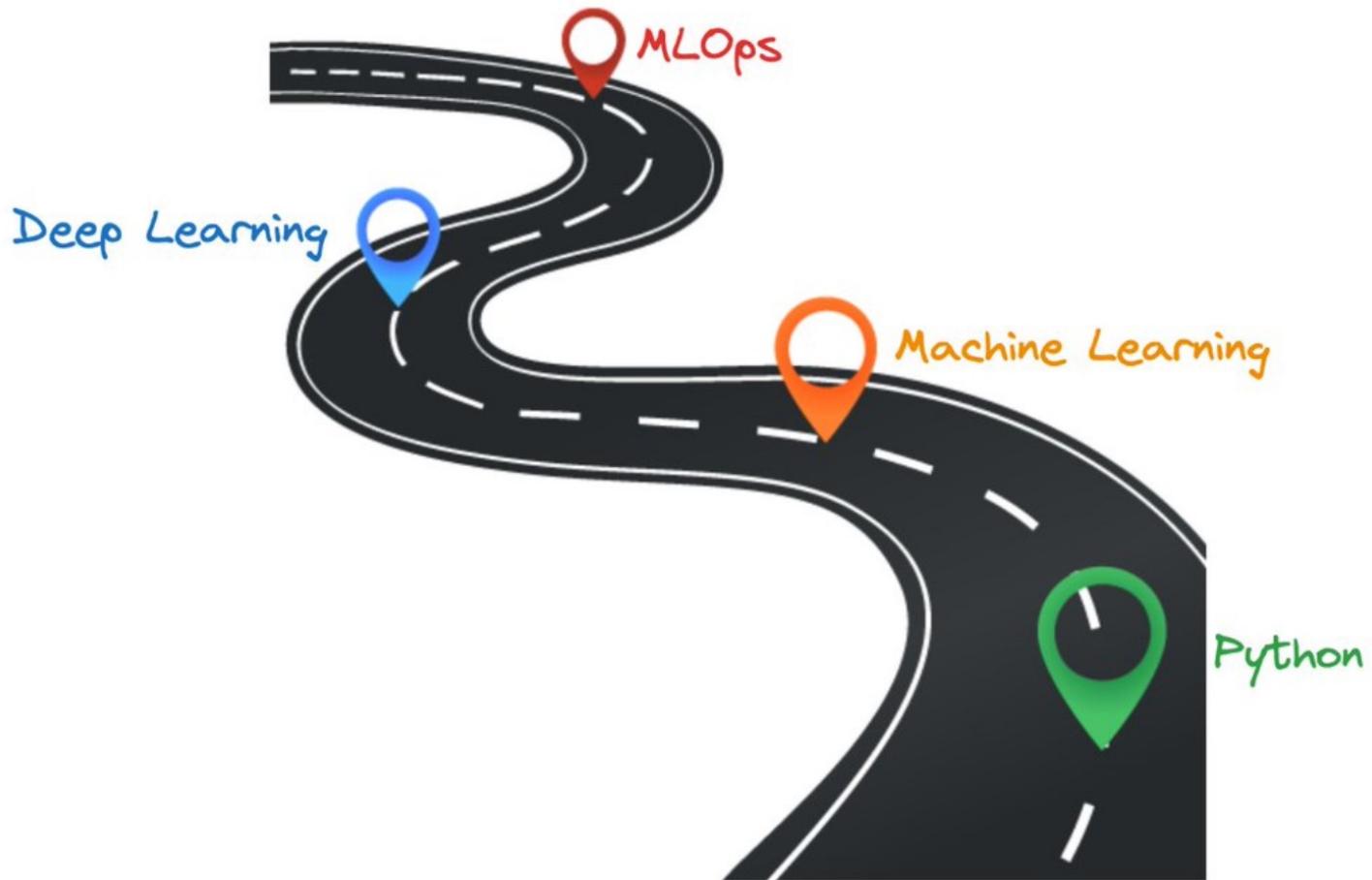
SUMMER SCHOOL



ARTIFICIAL INTELLIGENCE SUMMER SCHOOL 2023

Introduction to AI: Road map

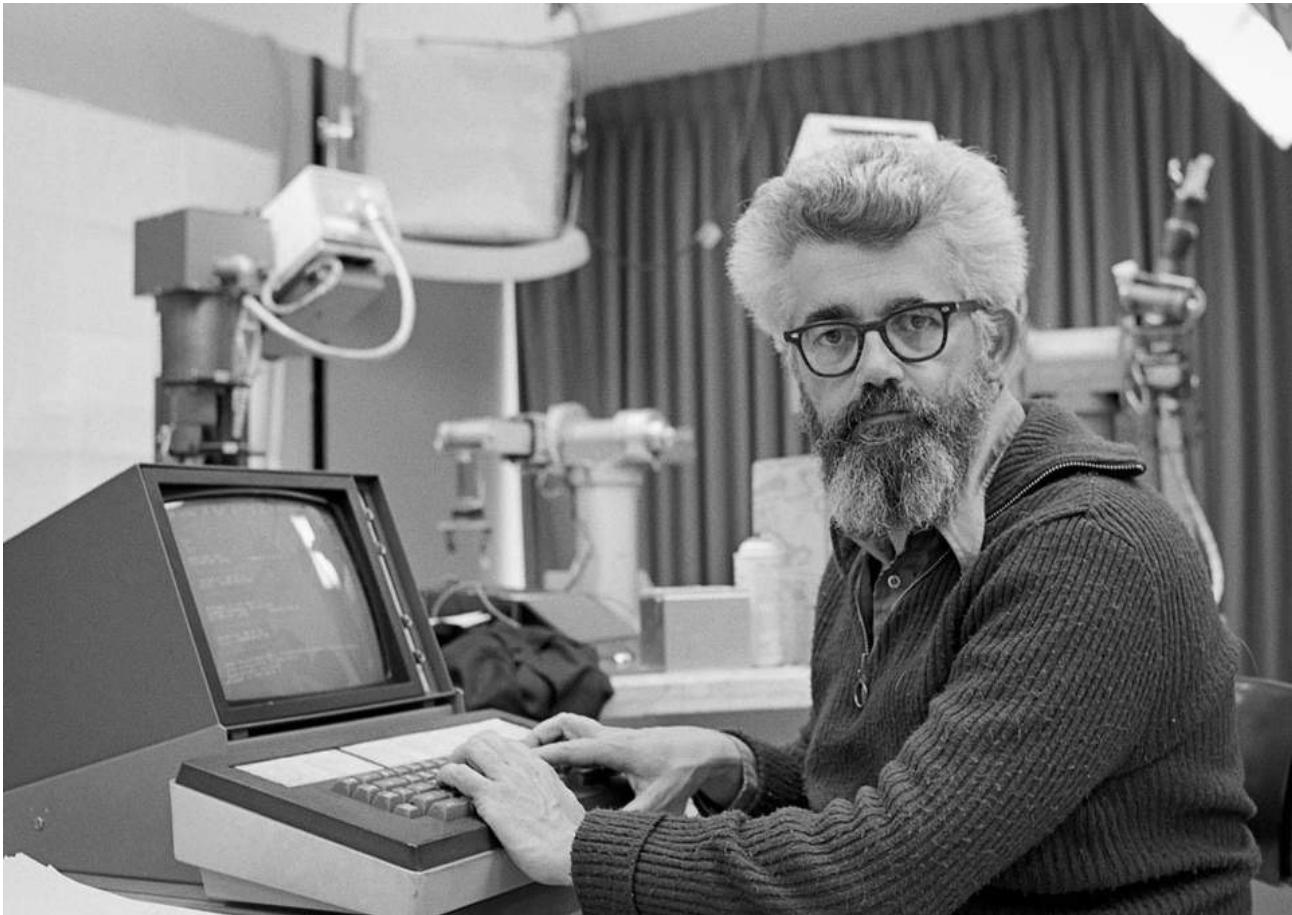
1. AI: a definition
2. AI and Machine learning
3. AI today
4. AI in cars
5. ChatGPT
6. Responsible AI



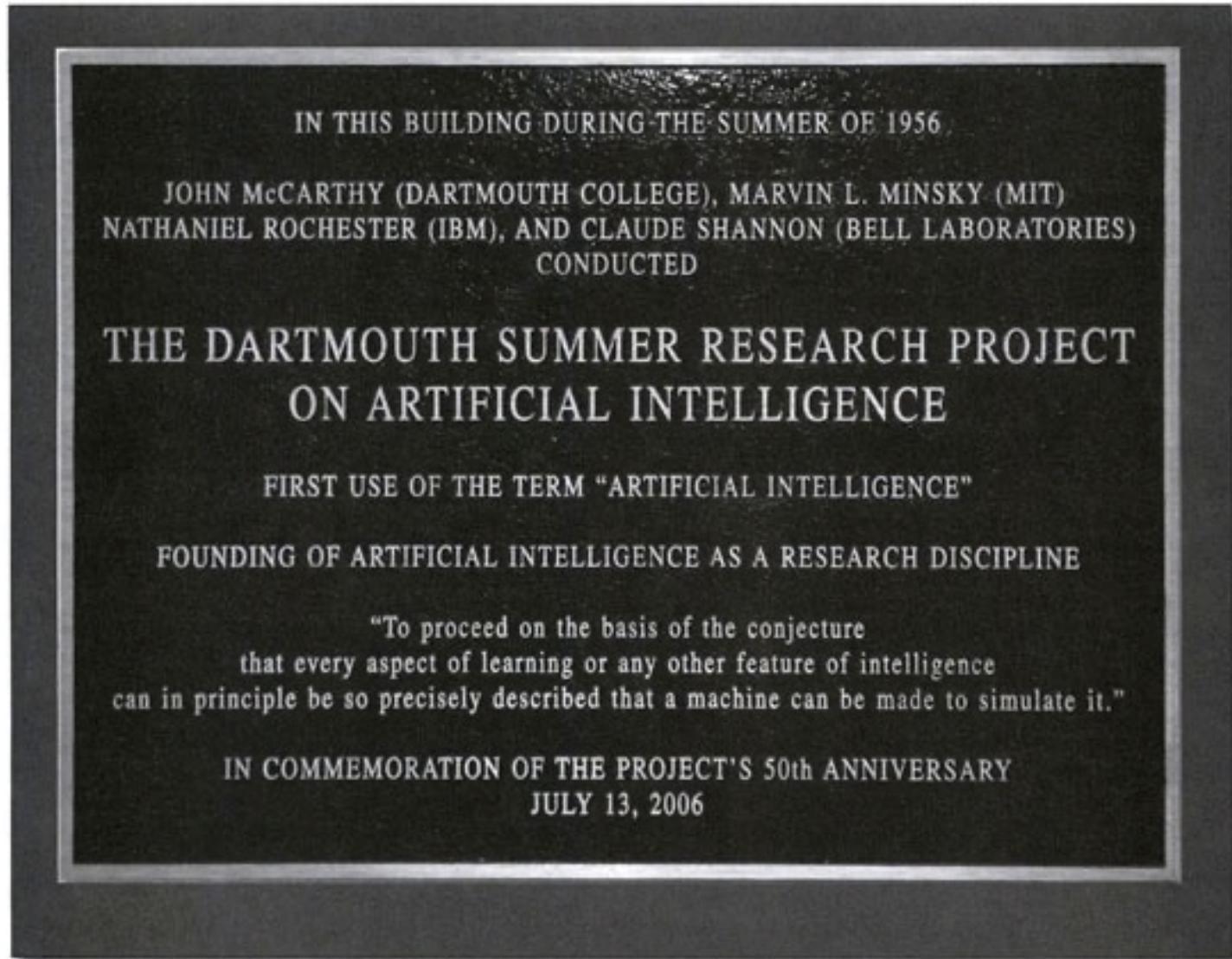
**HAPPY
NEW YEAR
1927**



1927: Year of birth of John McCarthy,
THE (grand) father of AI



1956: the Dartmouth Summer Research Project



To do's list # 1: define AI

The study is to proceed on the basis of the conjecture that every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it.

learning vs other feature of intelligence

learning vs other feature of intelligence

Intelligence

- A very general mental capability that, among other things, involves the ability to reason, plan, solve problems, think abstractly, comprehend complex ideas, **learn quickly and learn from experience.**

Learning for a child

- To walk (one year)
- To talk (two years)
- **To think** (rest of its life...)

Two visions of AI

learning vs other feature of intelligence: How to build an autonomous vehicle?

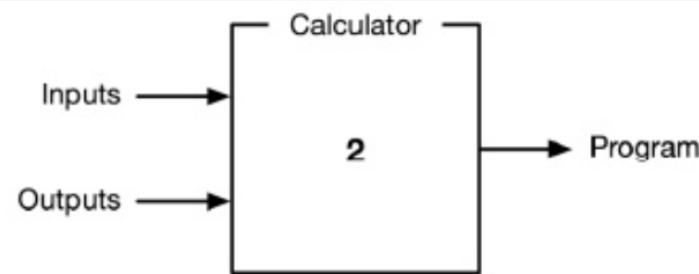
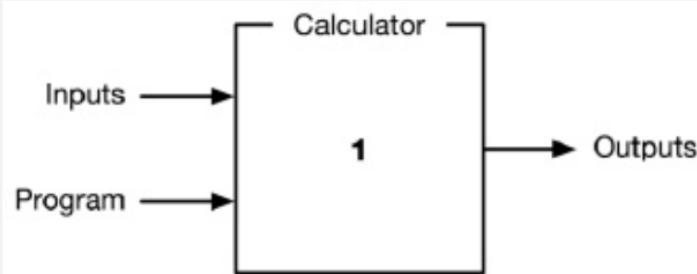
Building it:

- by modeling the driving process

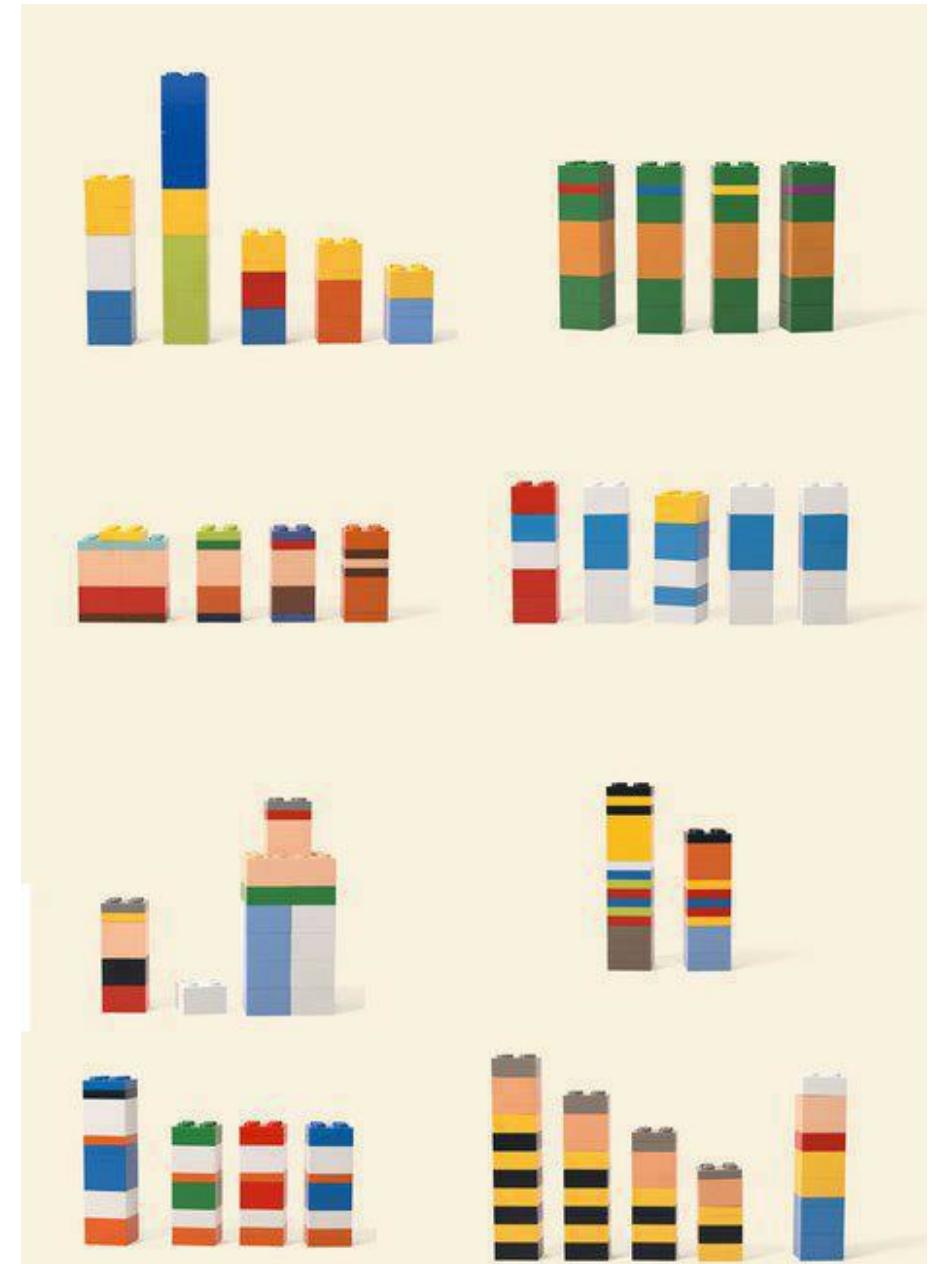
By learning:

- by imitating the driver

HYPOTHETICAL-DEDUCTIVE MACHINES (1) AND INDUCTIVE MACHINES (2)

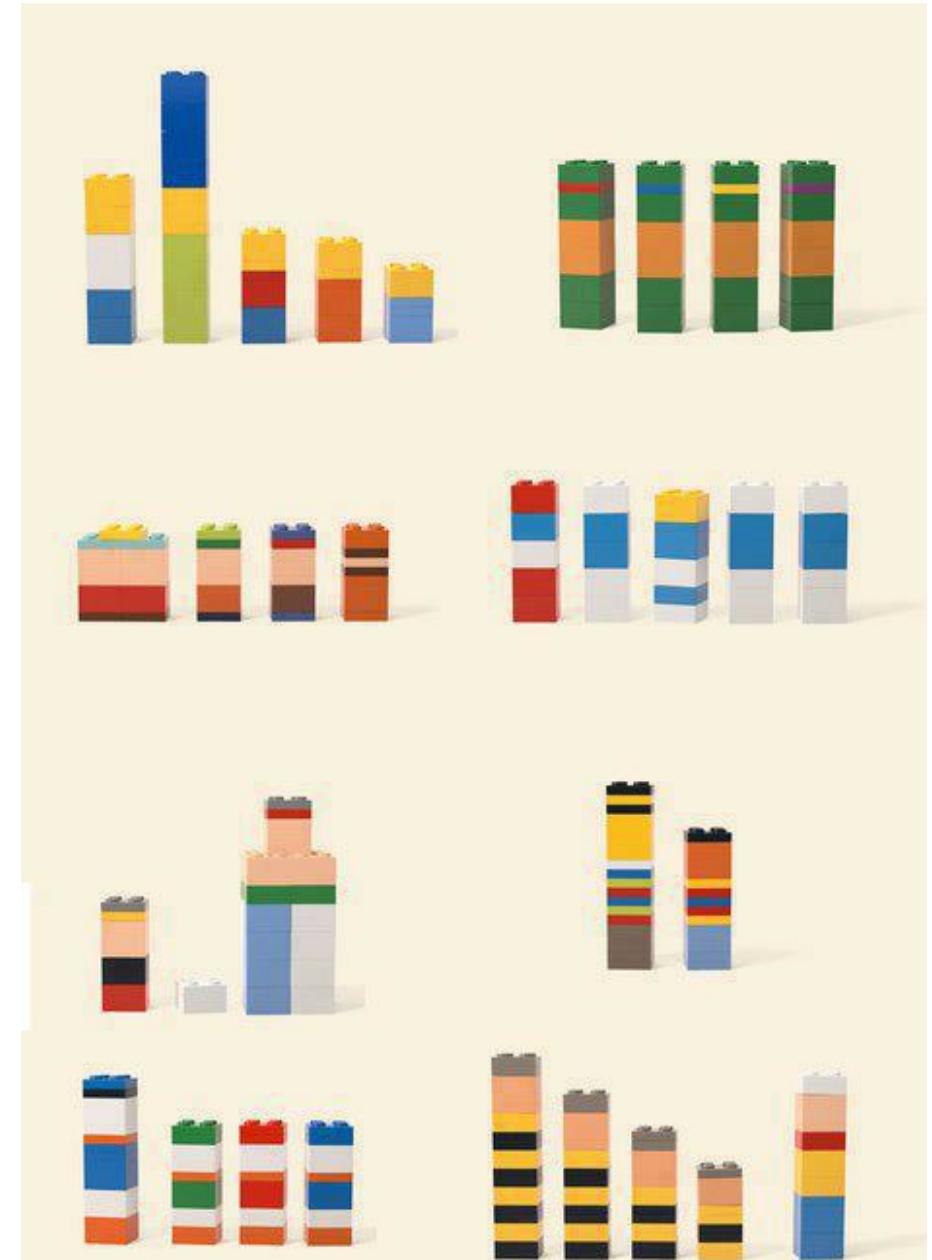


Example: learn to recognize objects



Example: learn to recognize objects

Context really matters!



3 different kinds of AI



What can you see (is it a cat?)

Specific AI (narrow)

Why is she a cat?

General AI

Is she going up or down?

Supper AI

3 different kinds of AI

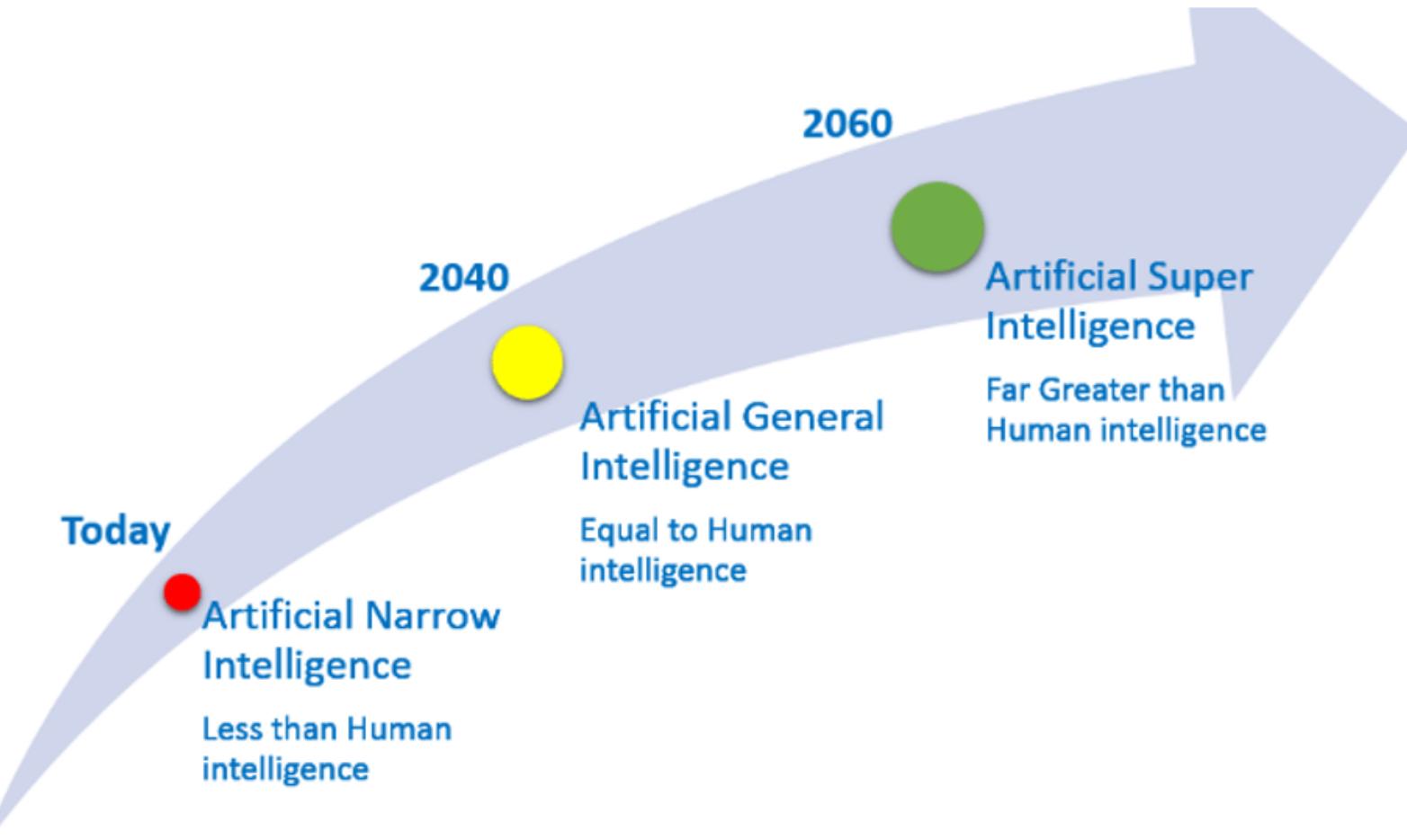


Figure 4: Future evolution of Artificial Intelligence

- <https://www.linkedin.com/pulse/ai-definitions-narrow-vs-general-super-intelligence-troy-angrignon/>

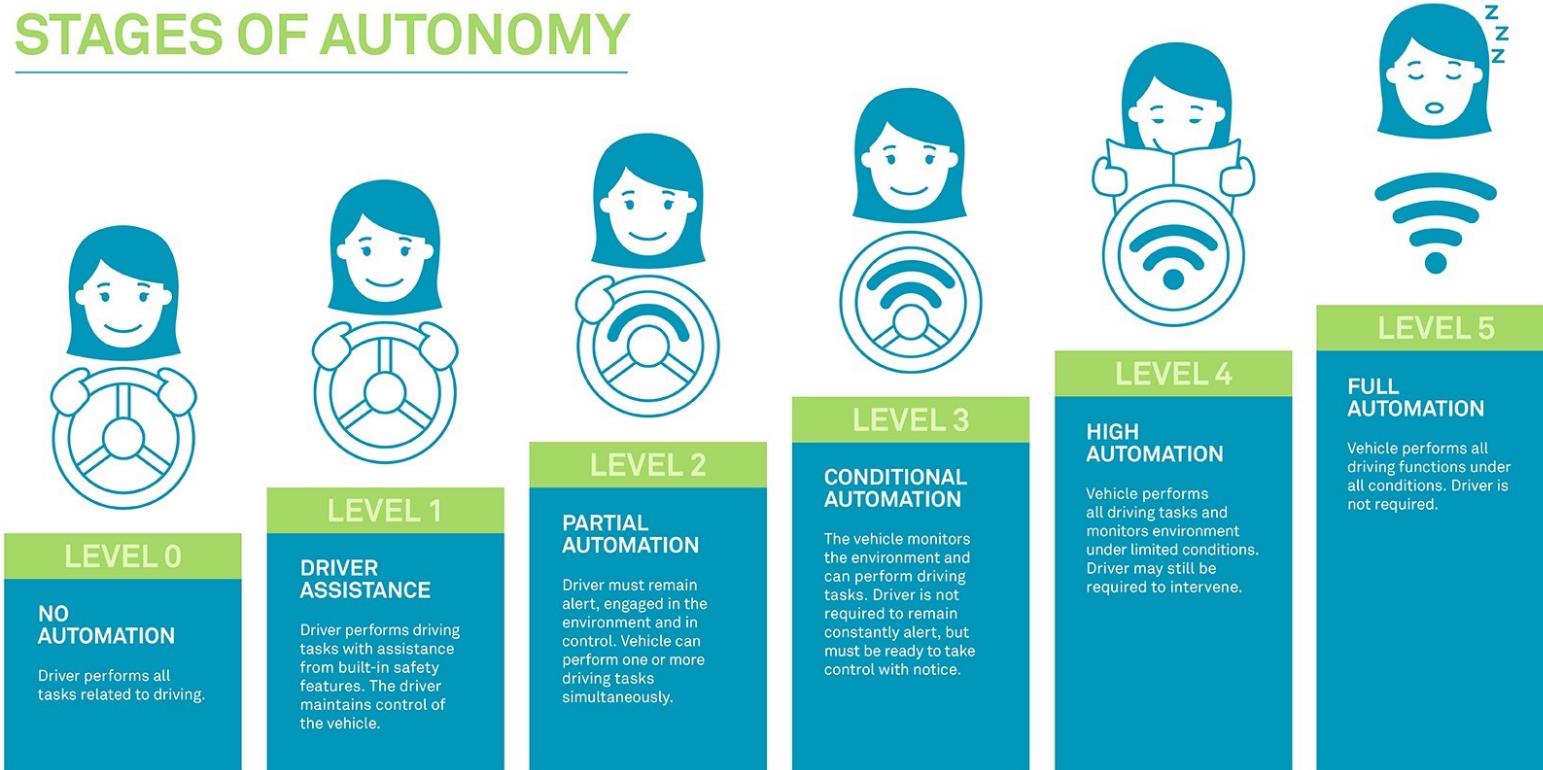
Super intelligence: when?



In a while, after some major breakthrough...

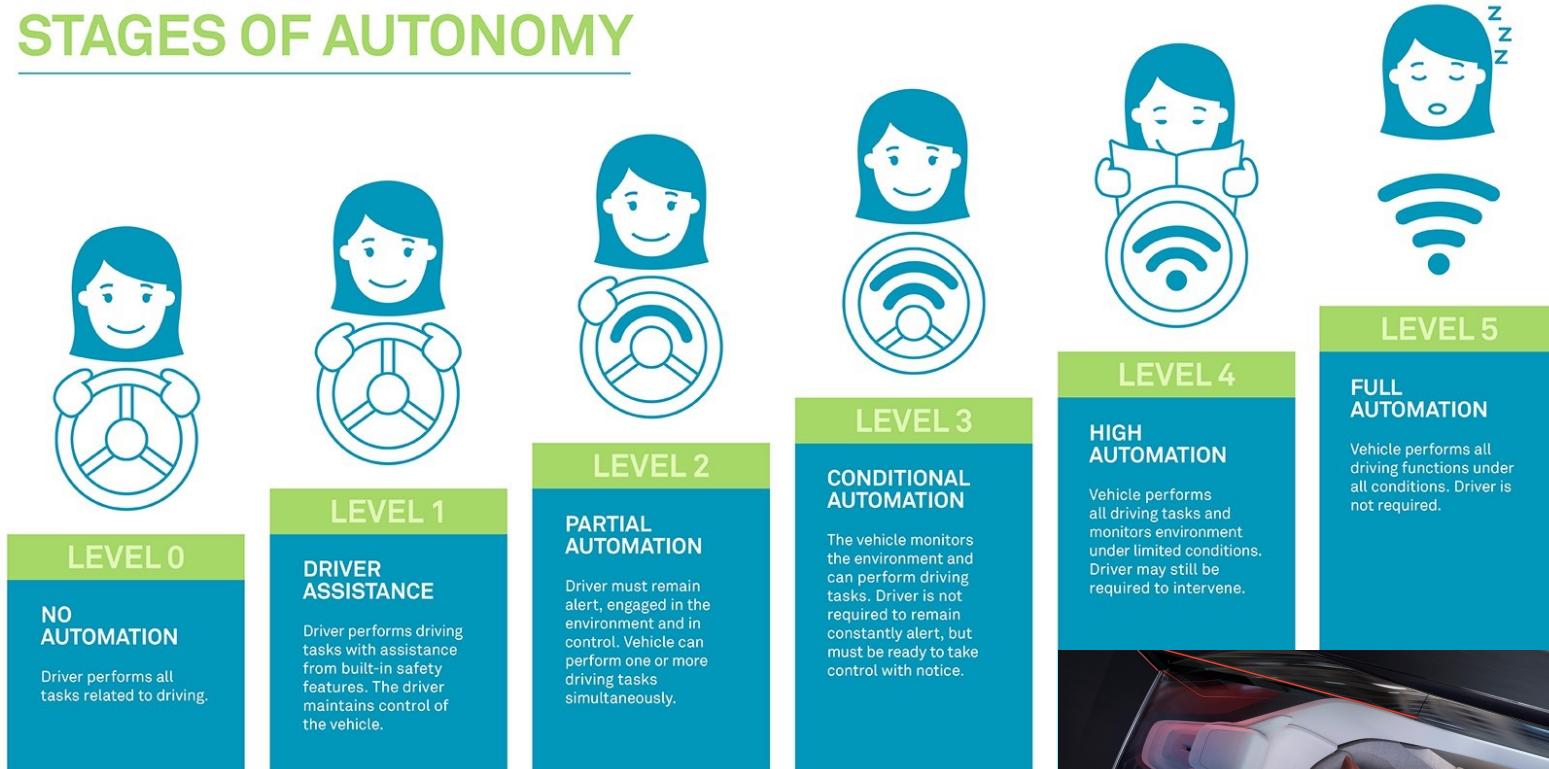
Autonomous driving

STAGES OF AUTONOMY



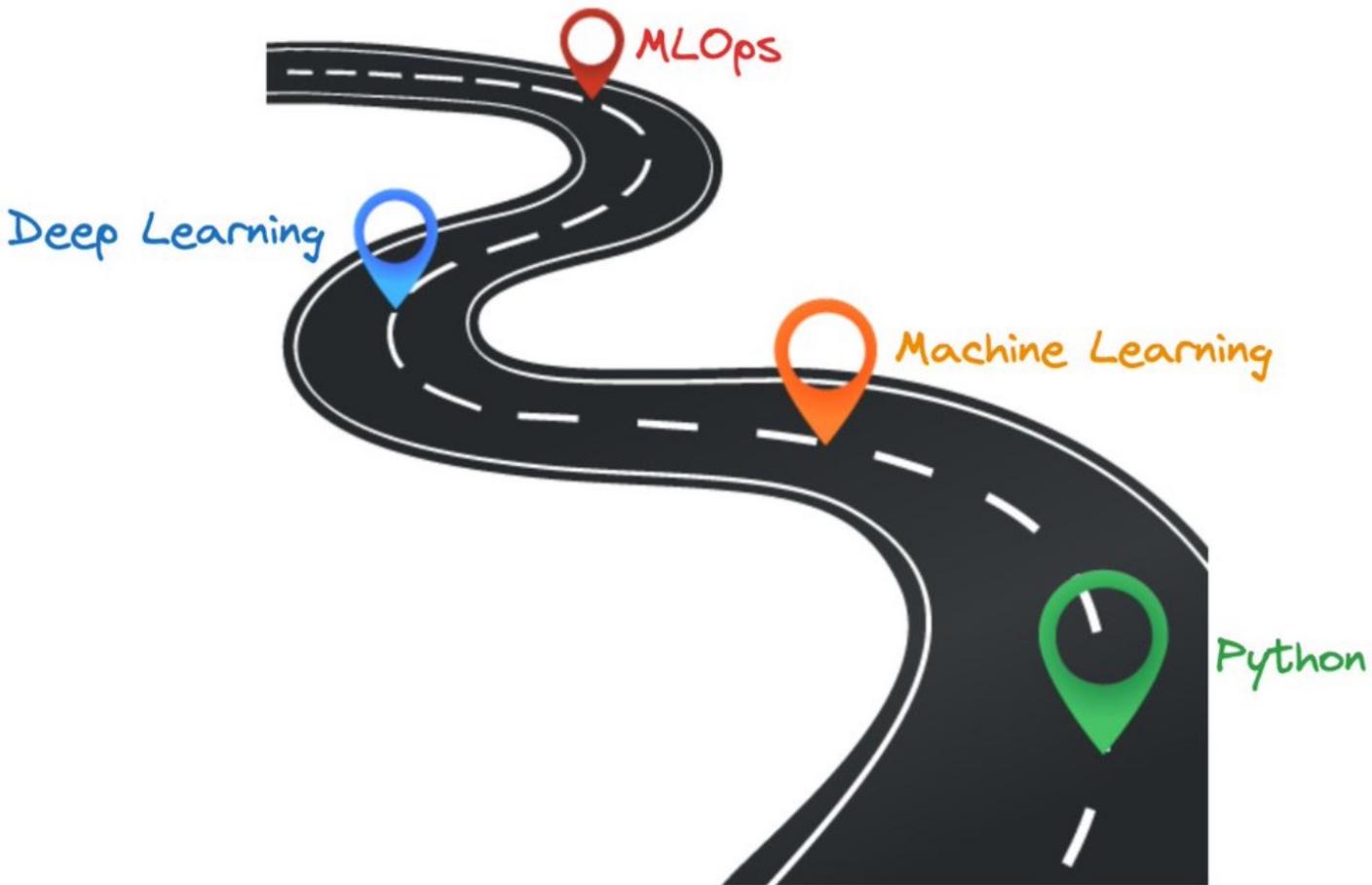
Autonomous driving

STAGES OF AUTONOMY



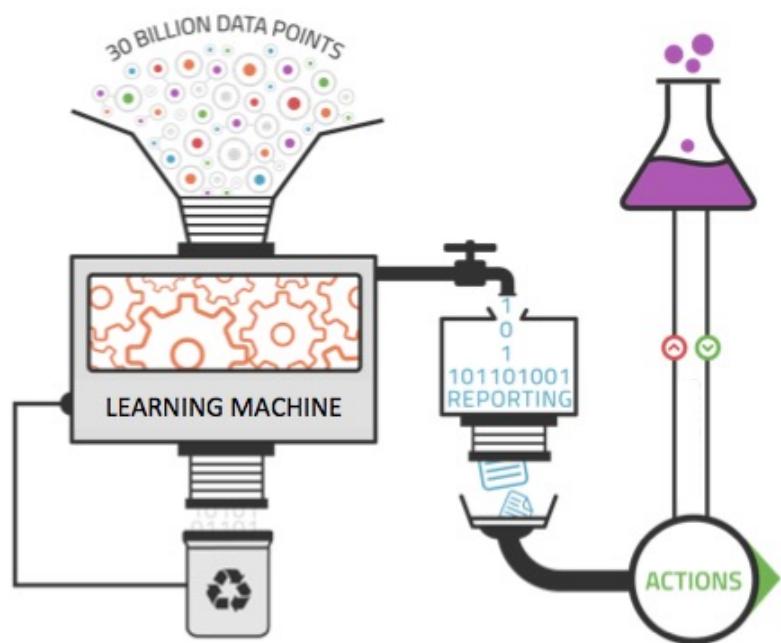
Introduction to AI: Road map

1. AI: a definition
2. AI and Machine learning
3. AI today
4. AI in cars
5. ChatGPT
6. Responsible AI



How to build an intelligent machine?

Machine learning



Data Based programming

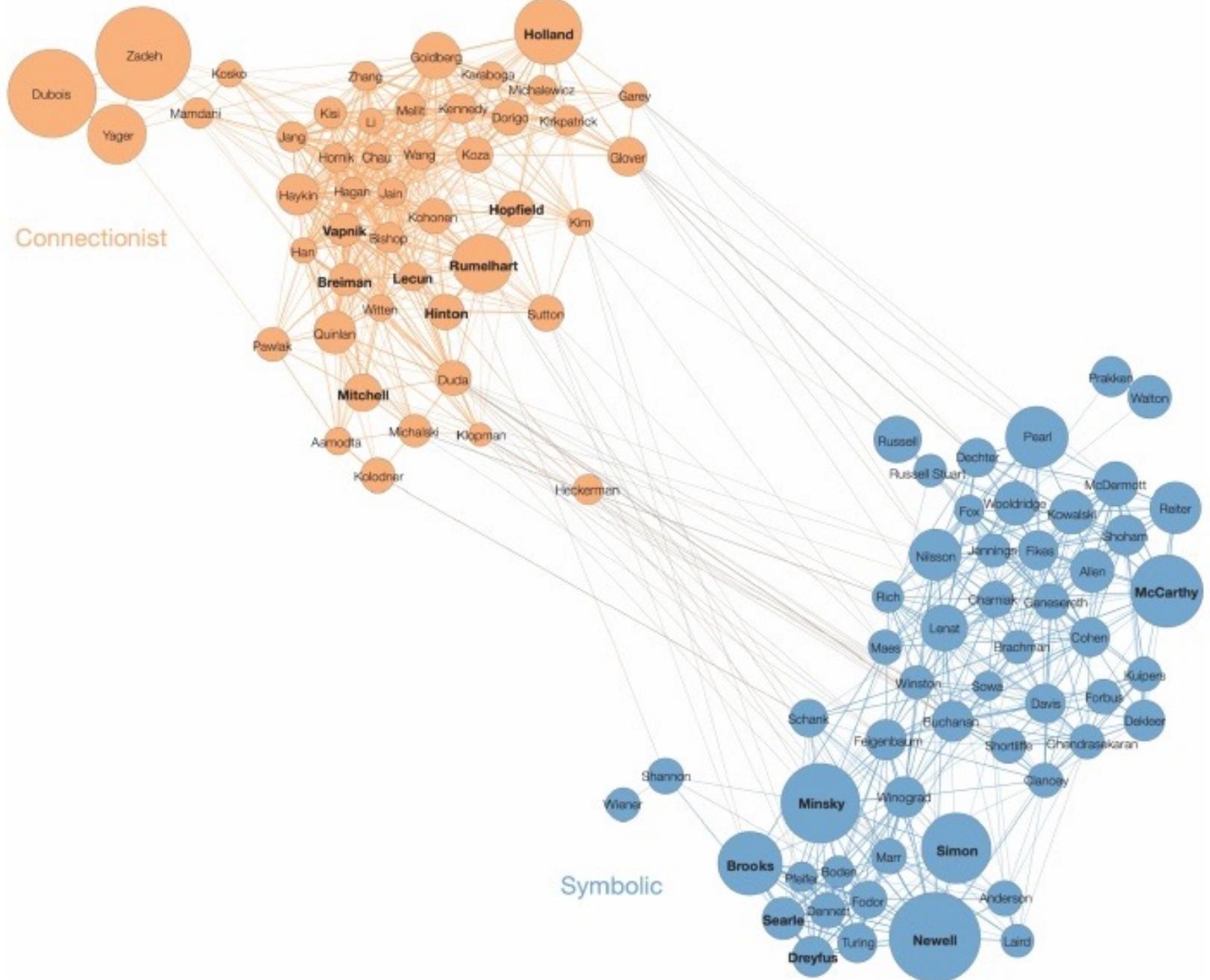
vs.

modeling reasoning



Knowledge based programming

Two different communities



learning vs other feature of intelligence: How to build an autonomous vehicle?

By learning:

- Driver imitation



Stanford

Building it:

- Model the driver



Carnegie Mellon

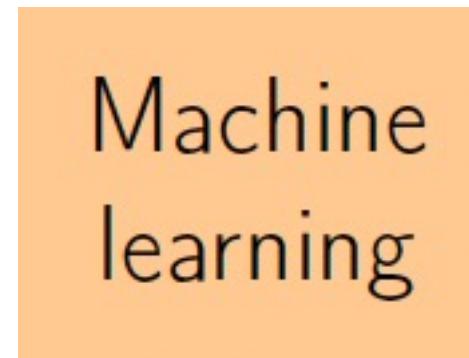
DARPA Grand Challenge 2005

2 million \$ – 132 miles (213 km) in the desert, Primm, Nevada

Machines learning recipe

- Inference
 - Goal (specific problem: recognize image, text, drive...)
 - Data (examples)
 - Hypothesis (prior knowledge)

Data :
Inputs = x
Outputs = y

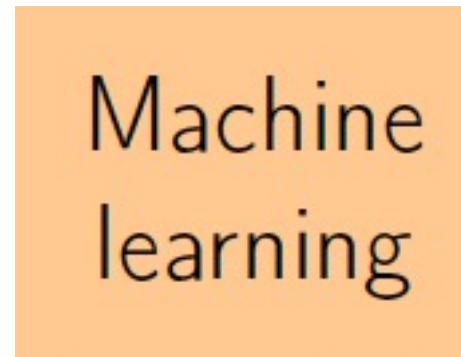


Model:
 f : input \xrightarrow{x} output
 $y = f(x)$

Machines learning recipe

- A computer program A learn from experience E with respect to some class of tasks T and performance measure L , if its performance at tasks in T , as measured by L , improves with experience E
- Inference
 - Goal $T & L$ (specific problem: recognize image, text, drive...)
 - Data E (examples)
 - Hypothesis A (prior knowledge)

Data :
Inputs = x
Outputs = y

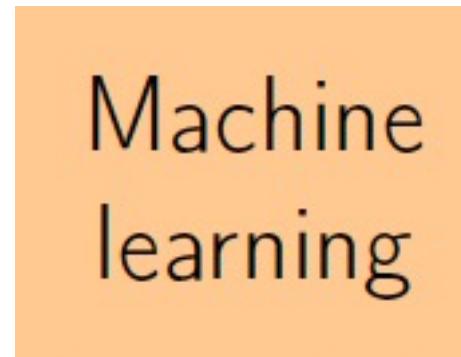


Model:
 f : input \xrightarrow{x} output
 $y = f(x)$

Machines learning recipe (T. Mitchell, 2006)

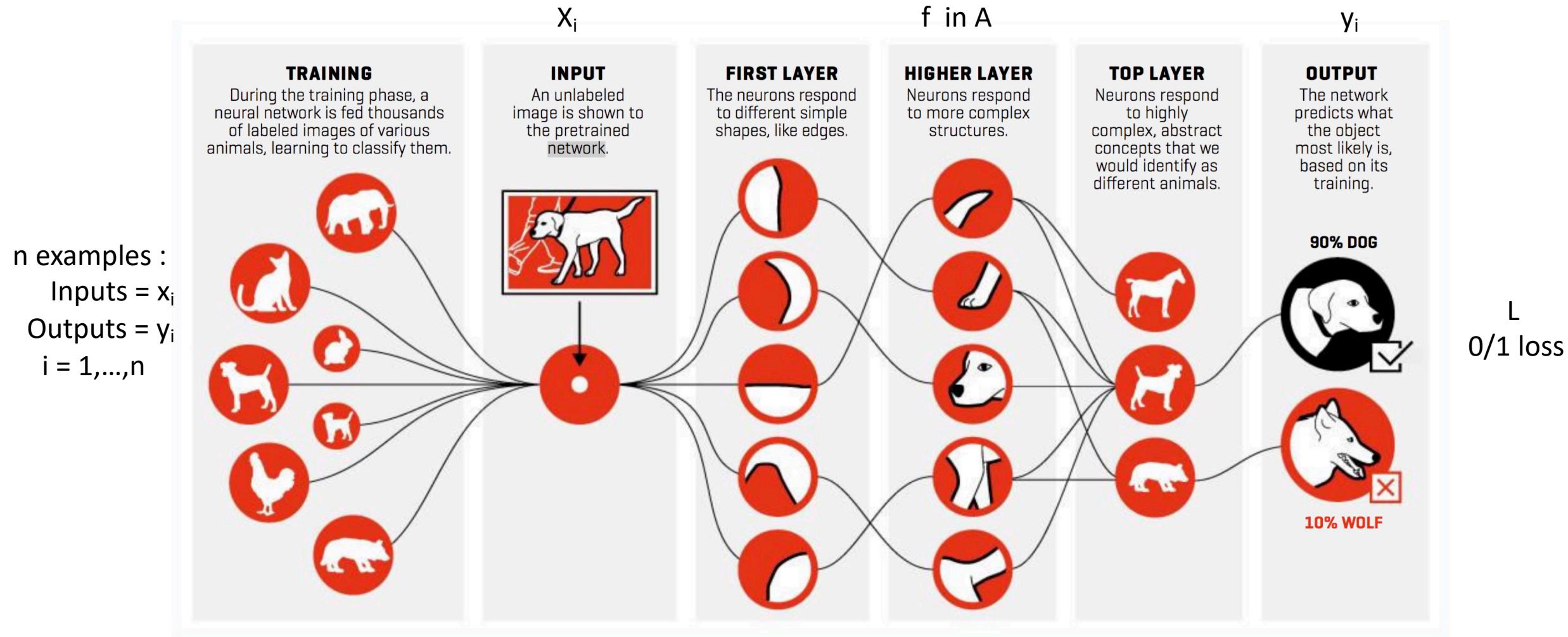
- A computer program A learn from $\text{experience } E$ with respect to $\text{some class of tasks } T$ and $\text{performance measure } L$, if its performance at $\text{tasks in } T$, as measured by L , improves with experience E
- Inference
 - Goal $T \& L$ recognize a cat in an image: $L = \text{error rate}$
 - Data E (n examples) a set of images of cat and non cats (dogs?)
 - Hypothesis A (prior knowledge) f is a neural network, SVM, random forest...

n examples :
Inputs = x_i
Outputs = y_i
 $i = 1, \dots, n$



$$f^* = \underset{f \text{ in } A}{\operatorname{argmin}} \sum_{i=1,n} L(f(x_i), y_i)$$

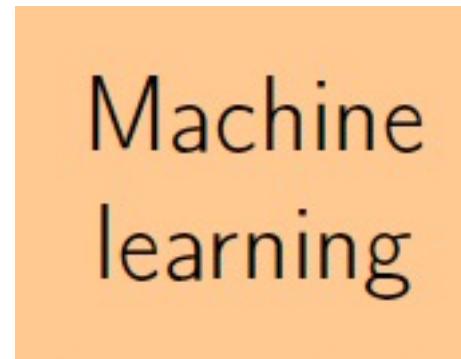
Learning machines: T = classification



Machine learning as an ill posed optimization problem (function estimation)

- Inference
 - Goal T & L
 - Data E (n examples)
 - Hypothesis A (prior knowledge)
- recognize a cat in an image: L = error rate
a set of images of cat and non cats (dogs?)
 f is a neural network, SVM, random forest...

n examples :
Inputs = x_i
Outputs = y_i
 $i = 1, \dots, n$



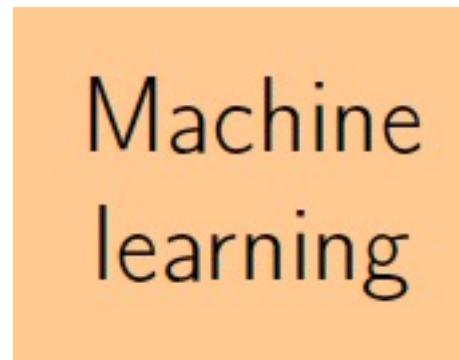
$$f^* = \underset{f \text{ in } A}{\operatorname{argmin}} \sum_{i=1,n} L(f(x_i), y_i)$$

Machine learning as an ill posed optimization problem (function estimation)

- Inference

- Goal T & L
 - Data E (n examples)
 - Hypothesis A (prior knowledge)
- recognize a cat in an image: L = error rate
a set of images of cat and non cats (dogs?)
 f is a neural network, SVM, random forest...

n examples :
Inputs = x_i
Outputs = y_i
 $i = 1, \dots, n$

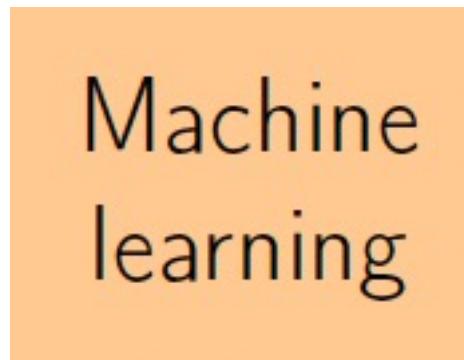


f^* that perform well on future data
= Generalization

$$f^* = \underset{f \text{ in } A}{\operatorname{argmin}} \sum_{i=1,n} L(f(x_i), y_i)$$
A hand-drawn style diagram consisting of two intersecting orange lines forming an 'X' shape, representing the search space for finding the optimal hypothesis f^* .

How to learn?

Data :
Inputs = x
Outputs = y



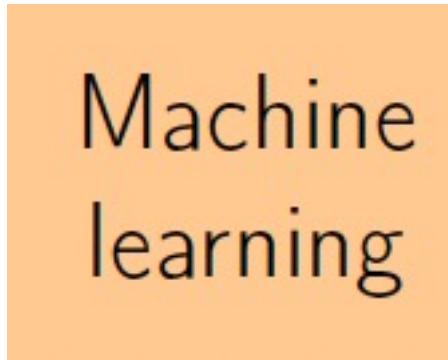
Model:

f: input output
 x \rightarrow $y = f(x)$

1. With a teacher (or a book)
 - Learn knowledge
2. From examples
 - Learn to reproduce a competency
 - To recognize a mushroom
 - Learn through trial and test
 - Learn to play chess or Go
 - Lear alone: learn to discover
 - To classify pictures

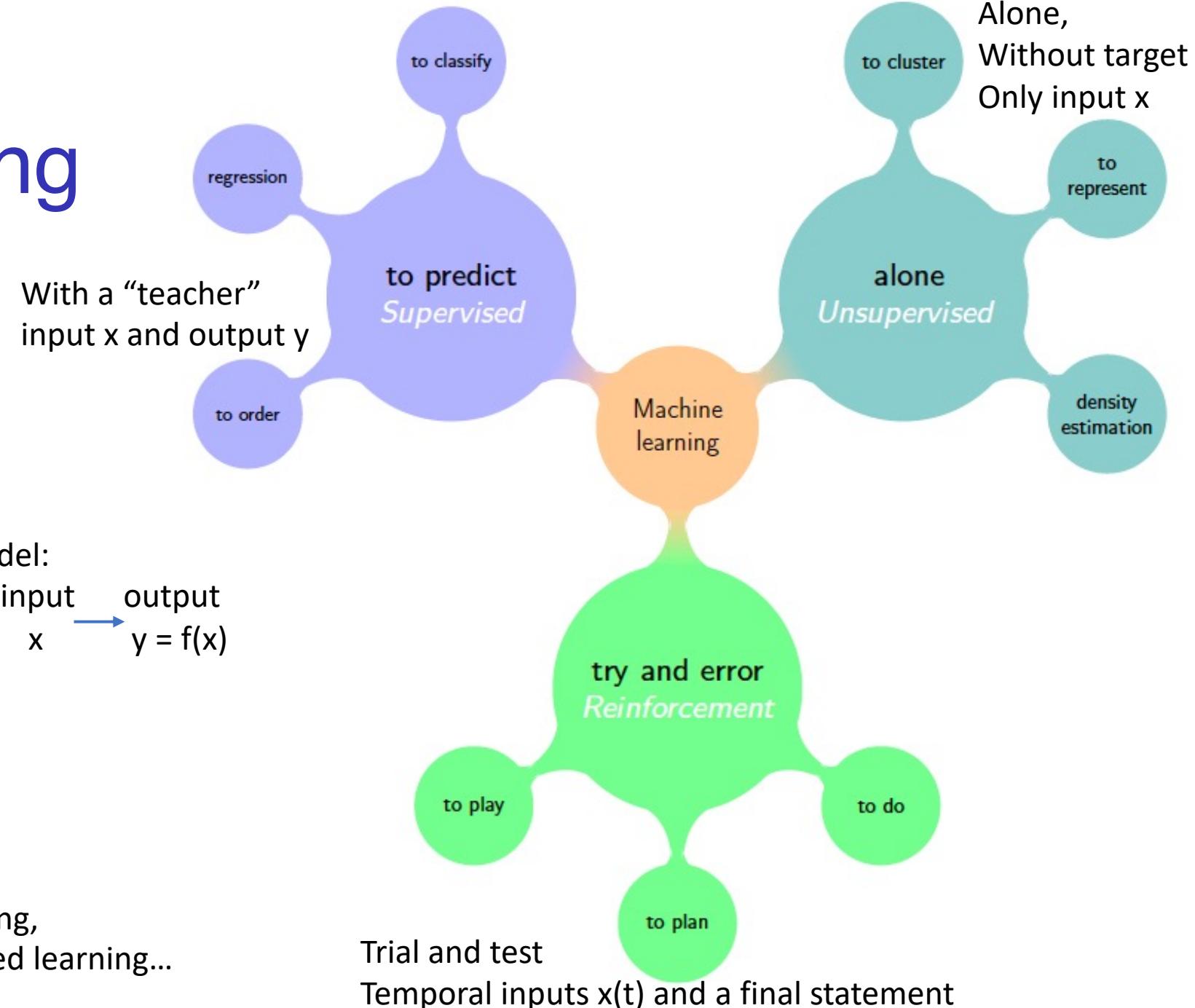
3 types of Machine learning

Data :
Inputs = x
Outputs = y



Model:
 $f: \text{input } x \rightarrow \text{output } y = f(x)$

And also: transfer learning, active learning,
online learning, semi supervised learning...



Supervised learning

input x and output y

- y is a real value (e.g. O₃ concentration...)
- y is a category (e.g. sick or healthy, object type...)
- y is a preference (e.g. between two explanations)
- y is a structured data (e.g. a graph, an image, a text...)



Main methods differs on the choice of the hypothesis: Neural networks, SVM, random forest, boosting...

Unsupervised learning

input x (no output, no y)

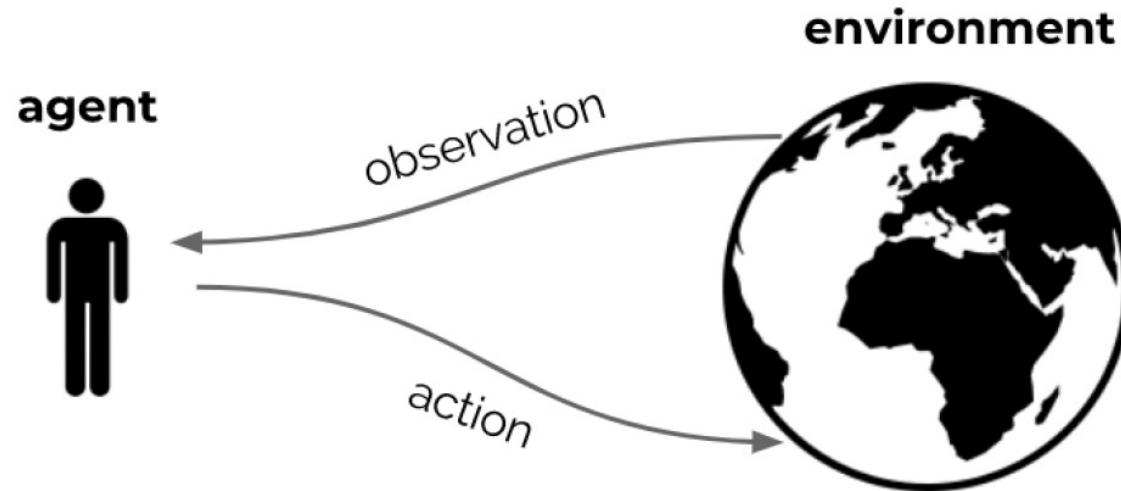
- Create categories (e.g. galaxy types...)
 - K-means, **DBSCAN**
- Represent (e.g. word to vec, vizualisation...)
 - PCA, **tSNE, UMAP**,
 - Self supervised: **deep learning** (ICLR)
- Density estimation (e.g. outlier detection, missing data...)

Could be seen a master problem in machine learning.
Solving this problem lets you solve a lot of other problems.
But hard to solve because of the curse of dimensionality...

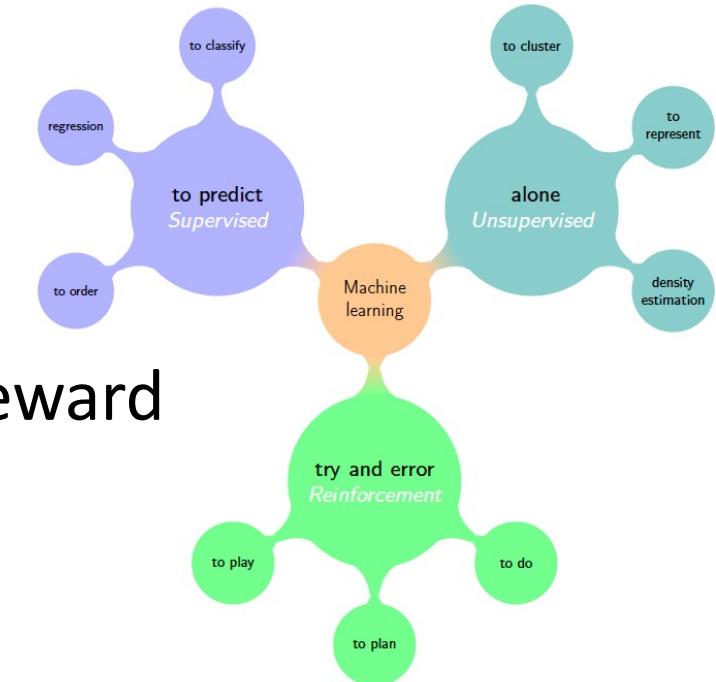


Reinforcement learning

Learning by interaction : Temporal input $x(t)$ and a final reward



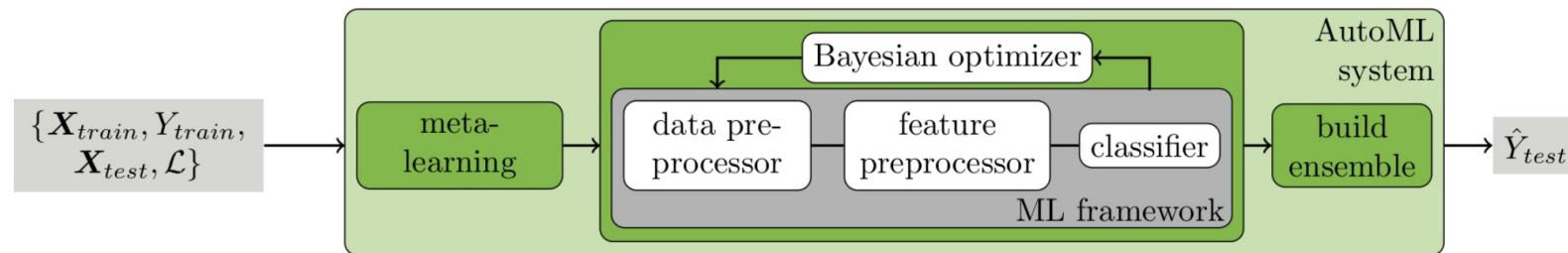
- Goal: maximizing a cumulative reward
 - e.g. win a game, maximize efficiency, the quality of a dialog (chatGPT)...
- Partially observable Markov decision problem (POMDP)
- Value based (Q-learning), policy based (PRO), Monte Carlo tree search, Actor-Critic RL...



Learning to learn

- AutoML: Automated machine learning
 - Use known problems (data set with the best ML solution)
 - Find similar problems to make suggestions
 - Train many with time budget (e.g. successive halving)
 - Model aggregation (stacking)

auto-sklearn in one image



auto-sklearn in four lines of code

```
import autosklearn.classification
cls = autosklearn.classification.AutoSklearnClassifier()
cls.fit(X_train, y_train)
predictions = cls.predict(X_test)
```

H₂O.ai



AutoGluon

github.com/automl/auto-sklearn

Kaggle: machine learning as a competition

[kaggle](#) Competitions Datasets Models Code Discussions Courses ...

Search

Inside Kaggle you'll find all the code & data you need to do your data science work. Use over 50,000 public datasets and 400,000 public notebooks to conquer any analysis in no time.

Maintained by Kaggle Starter Code Finance Datasets Linguistics Datasets Data Visualization Kernels

Financial Tweets David Wallach

Face Detection in Images DataTurks

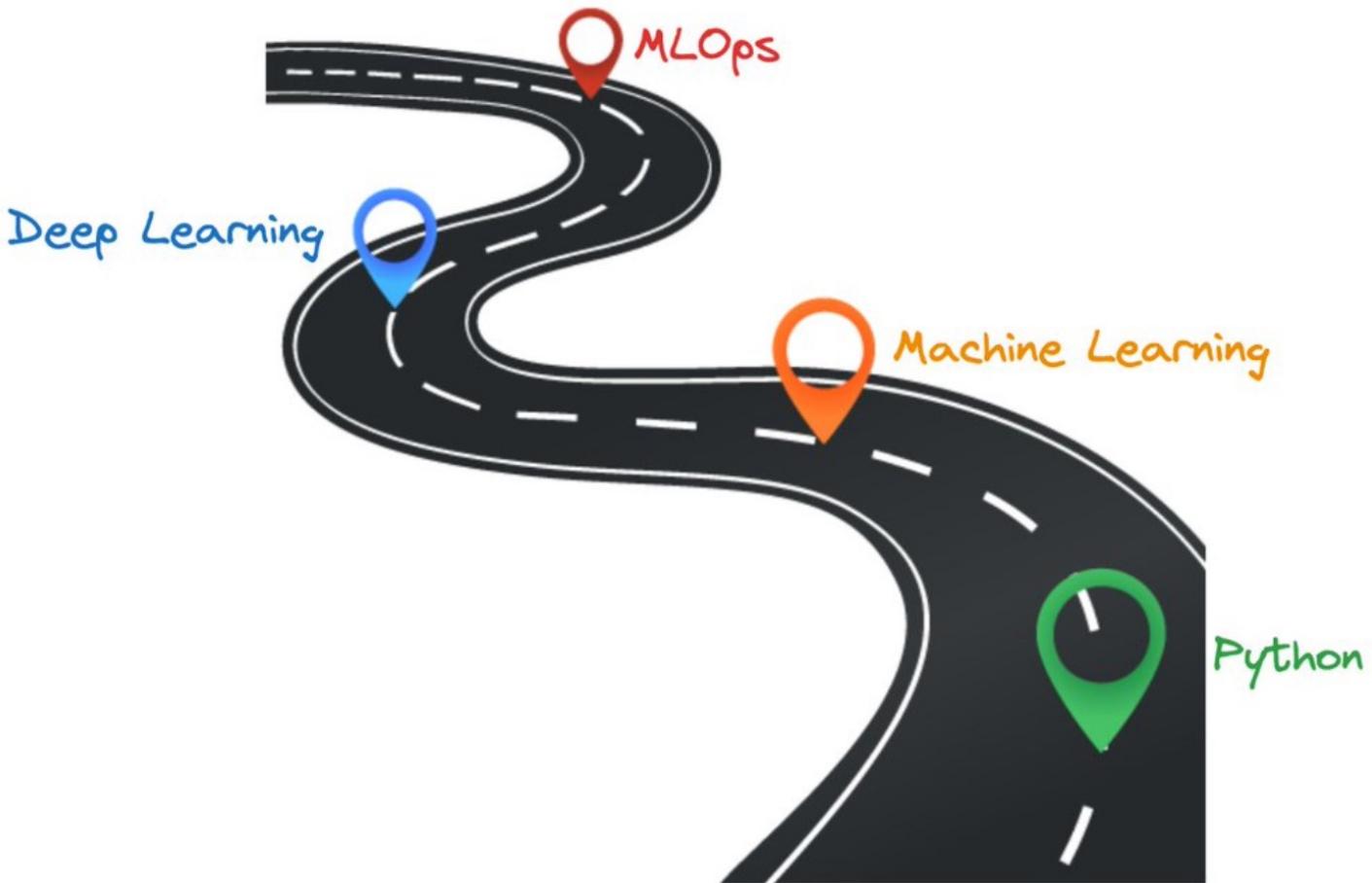
Star Trek Scripts Gary Broughton

Avocado Prices Justin Kiggins

Best methods: Deep learning, Boosting, random forest, SVM, AutoML + best practice (framework, encoding, tuning, Mlops...)

Introduction to AI: Road map

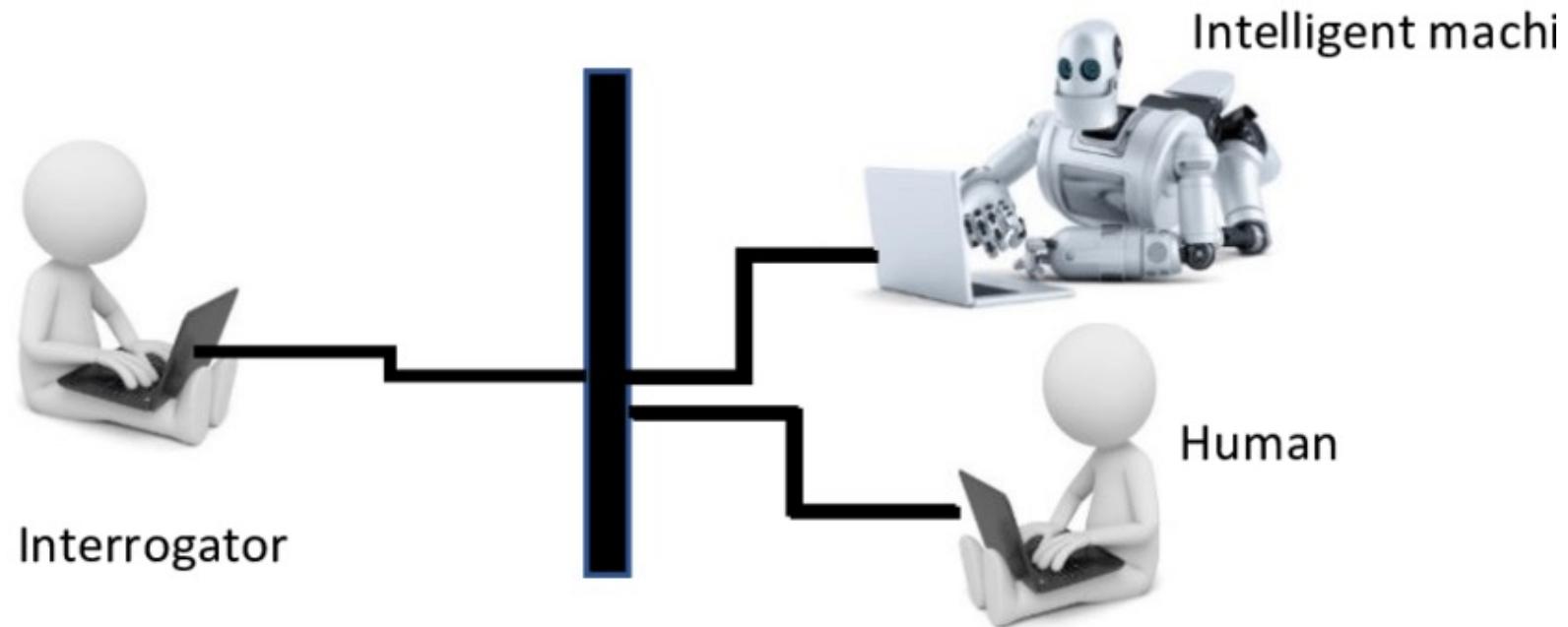
1. AI: a definition
2. AI and Machine learning
3. **AI today**
4. AI in cars
5. ChatGPT
6. Responsible AI



AI today: existing AI are specific

- How to recognize objects
- How to drive: **advanced driver-assistance system (ADAS)**
- How to make a medical diagnosis
- How to play games
- Speech to text
- Question answering – generating natural language (chatGPT)
- Information retrieval (chatGPT)
- ...

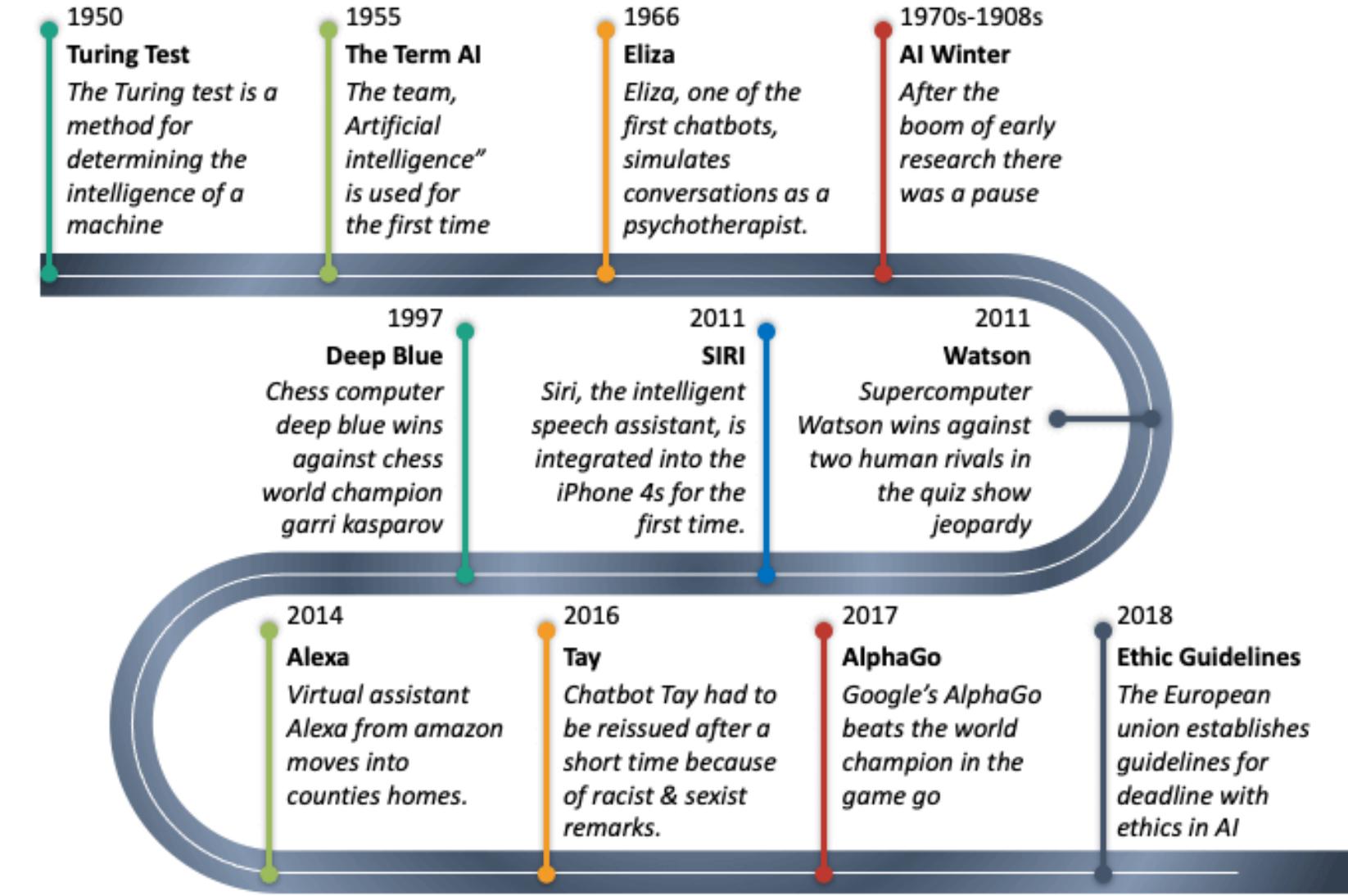
Turing test (the imitation game)



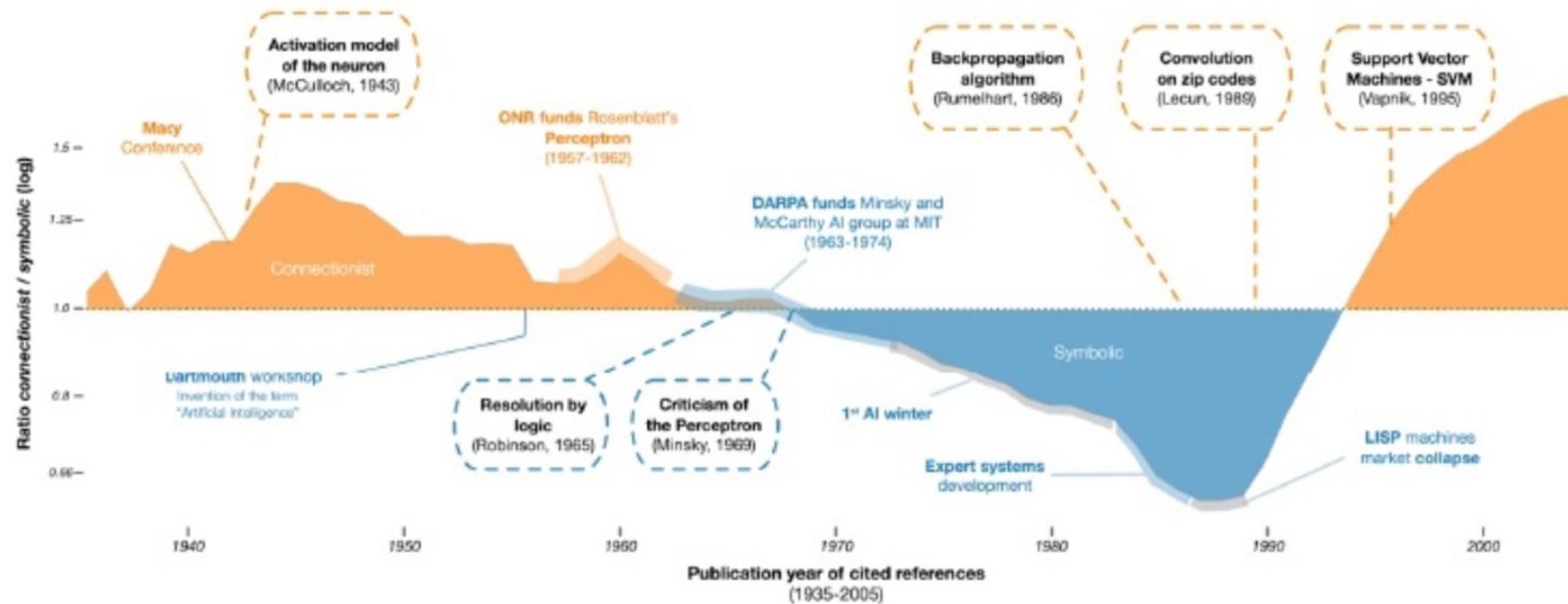
- https://en.wikipedia.org/wiki/Turing_test

AI time line

HISTORY OF ARTIFICIAL INTELLIGENCE (AI)



AI time line (and fashions)



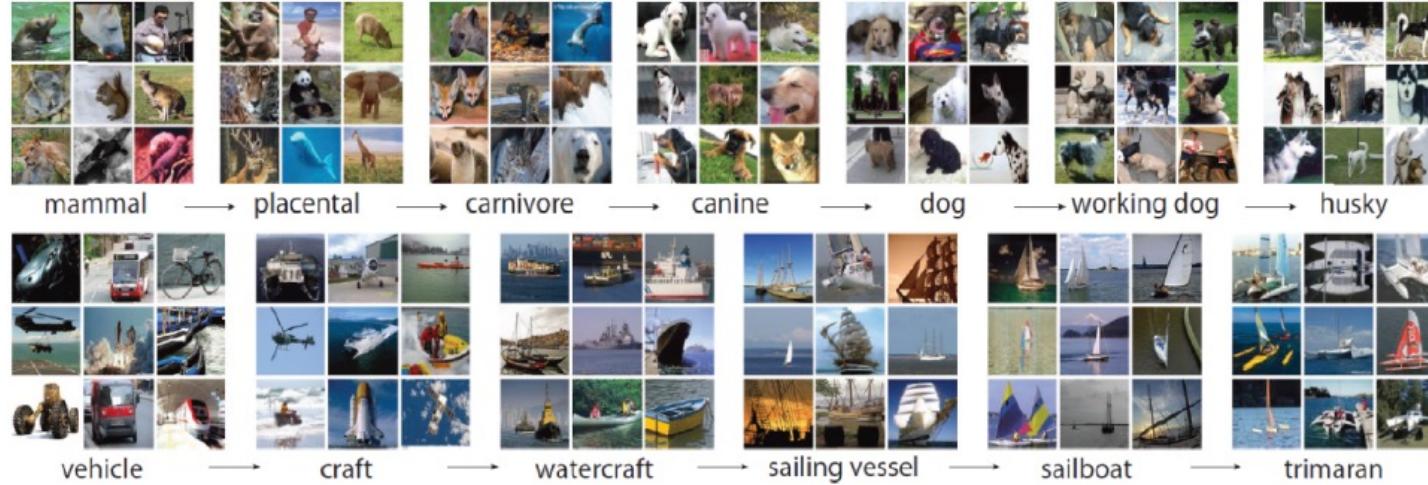
AI breakthroughs, useful, but not always as good as us humans

- 2012: How to recognize objects
 - ImageNet competition
- 2016: How to play games
 - AlphaGO
- 2022: Question answering – generating natural language
 - GPT -> chatGPT

Specific AI using a particular machine learning tool: **deep learning**

AI breakthroughs

2012: How to recognize objects database



ImageNet = 15 million high-resolution images of 22,000 categories.
Large-Scale Visual Recognition Challenge (a subset of ImageNet)

- 1000 categories.
- 1.2 million training images,
- 50,000 validation images,
- 150,000 testing images.

AI breakthroughs

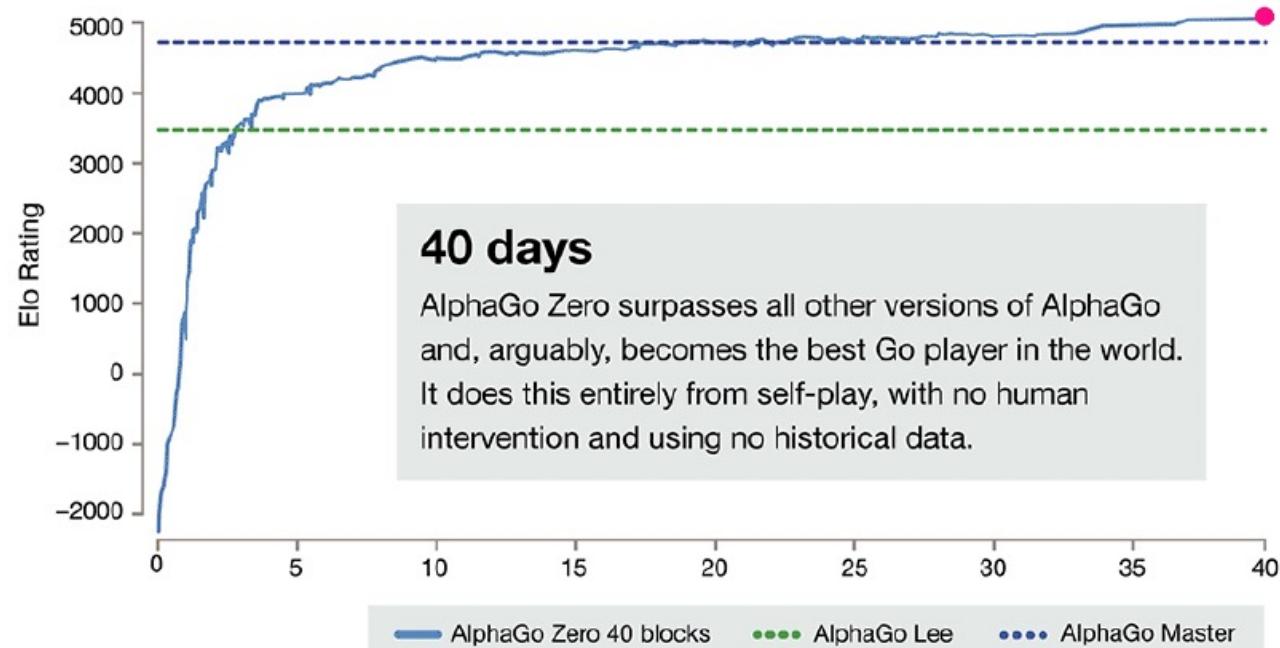
2012: How to recognize objects competition

2012 Teams	%error	2013 Teams	%error	2014 Teams	%error
Supervision (Toronto)	15.3	Clarifai (NYU spinoff)	11.7	GoogLeNet	6.6
ISI (Tokyo)	26.1	NUS (singapore)	12.9	VGG (Oxford)	7.3
VGG (Oxford)	26.9	Zeiler-Fergus (NYU)	13.5	MSRA	8.0
XRCE/INRIA	27.0	A. Howard	13.5	A. Howard	8.1
UvA (Amsterdam)	29.6	OverFeat (NYU)	14.1	DeeperVision	9.5
INRIA/LEAR	33.4	UvA (Amsterdam)	14.2	NUS-BST	9.7
		Adobe	15.2	TTIC-ECP	10.2
		VGG (Oxford)	15.2	XYZ	11.2
		VGG (Oxford)	23.0	UvA	12.1

 shallow approaches

 deep learning

AI breakthroughs 2016: How to play games



AI breakthroughs

2022: ChatGPT
how to answer questions?

200 million users in 2 months

what is chatGPT?

ChatGPT is an advanced language model developed by OpenAI. It belongs to the GPT (Generative Pre-trained Transformer) family of models and is specifically designed for generating human-like text responses in a conversational manner. It has been trained on a massive amount of text data from various sources to develop a deep understanding of language patterns, grammar, context, and even some level of common sense reasoning.

ChatGPT can understand and generate text based on the given prompt or input. It can carry on coherent and contextually relevant conversations, answer questions, provide explanations, give suggestions, and engage in interactive dialogue with users. It can be utilized in a wide range of applications, such as customer support, language translation, content generation, and educational tools.

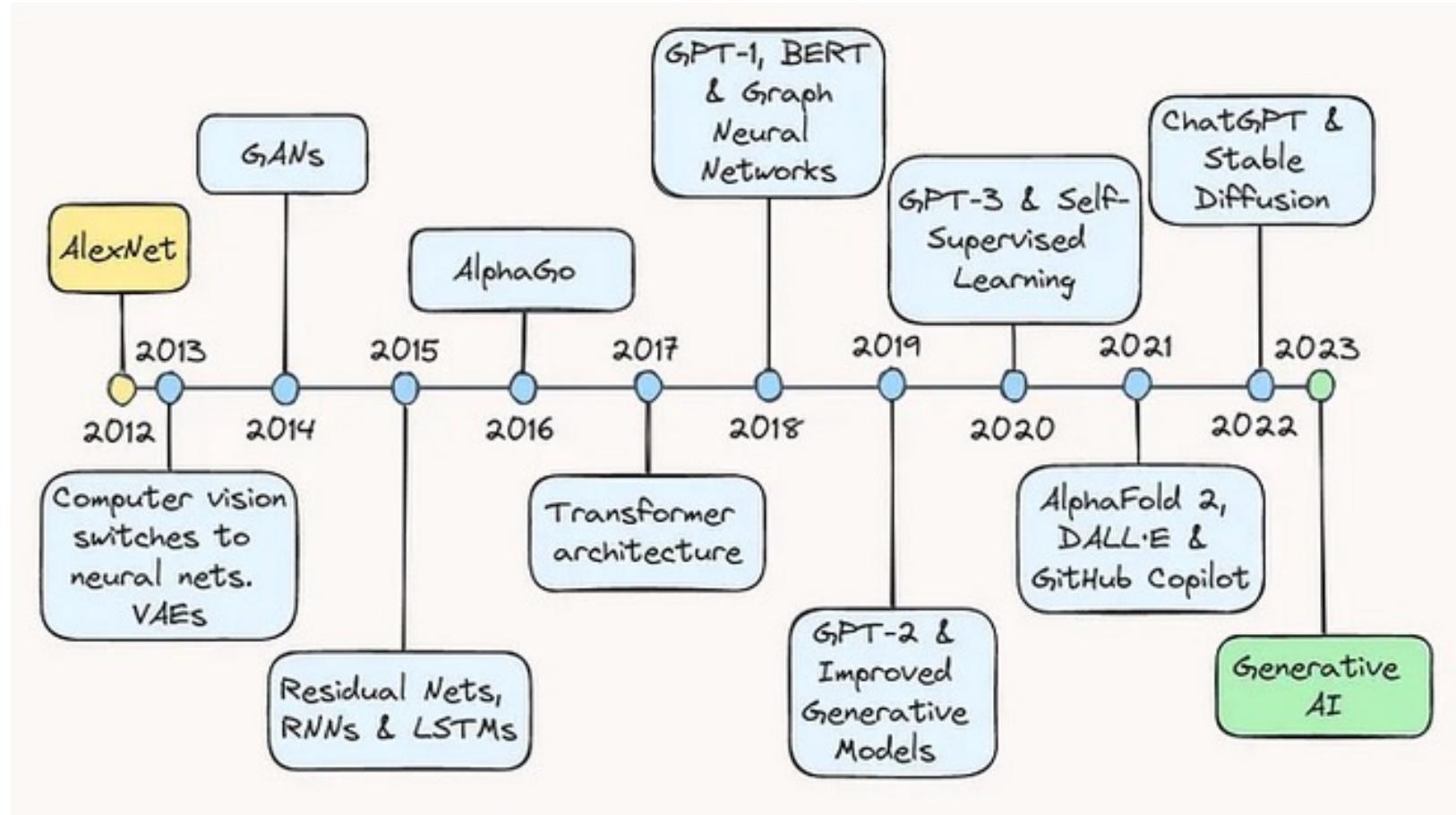
The model's responses are generated based on patterns and knowledge it has learned during training, but it's important to note that ChatGPT does not possess true understanding or consciousness. It generates responses purely based on statistical patterns in the training data and lacks real-time awareness or access to current information beyond its knowledge cutoff date.

Regenerate response

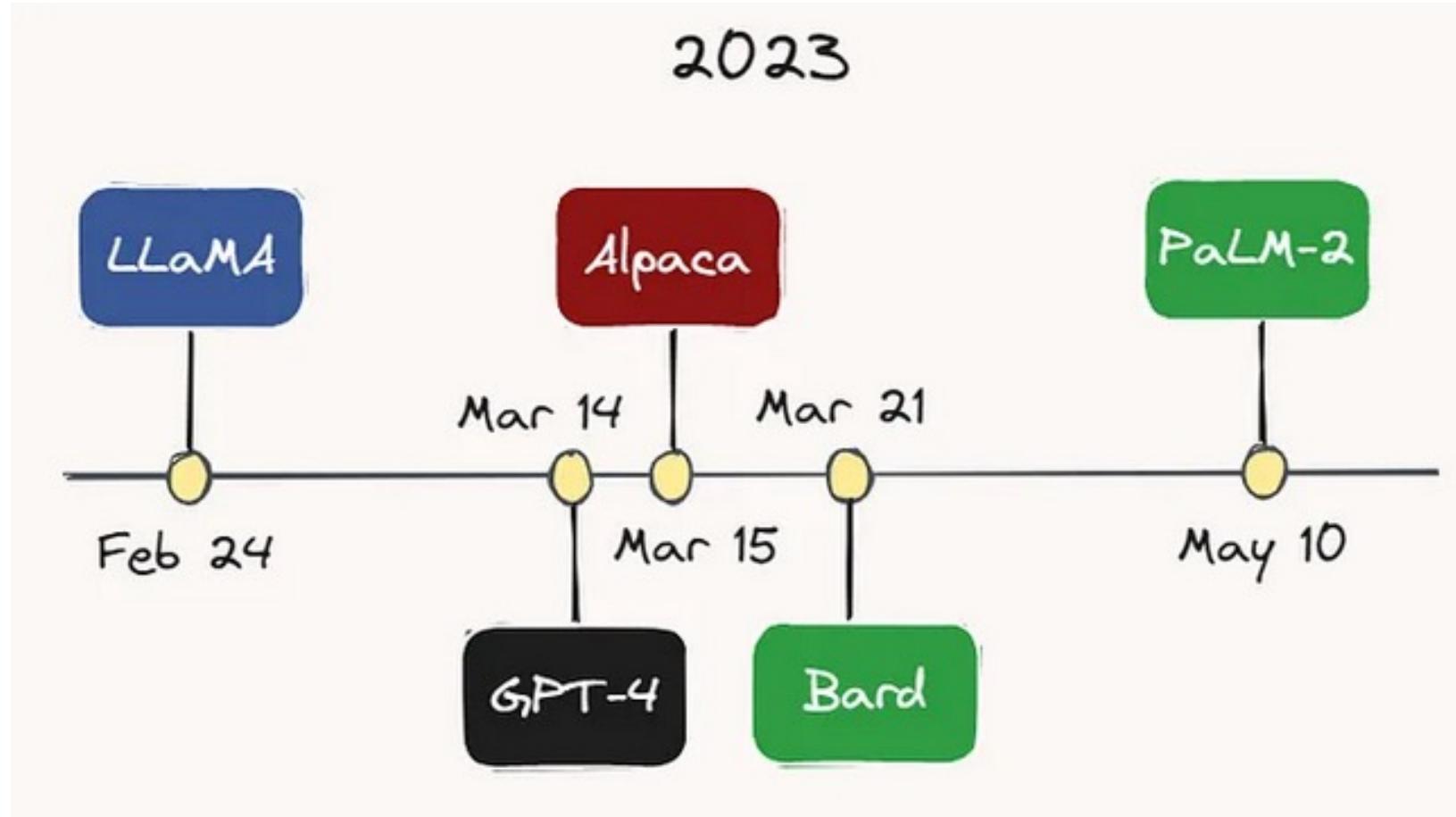
Send a message

Free Research Preview. ChatGPT may produce inaccurate information about people, places, or facts. ChatGPT May 24 Version

AI breakthroughs, 10 years in a review

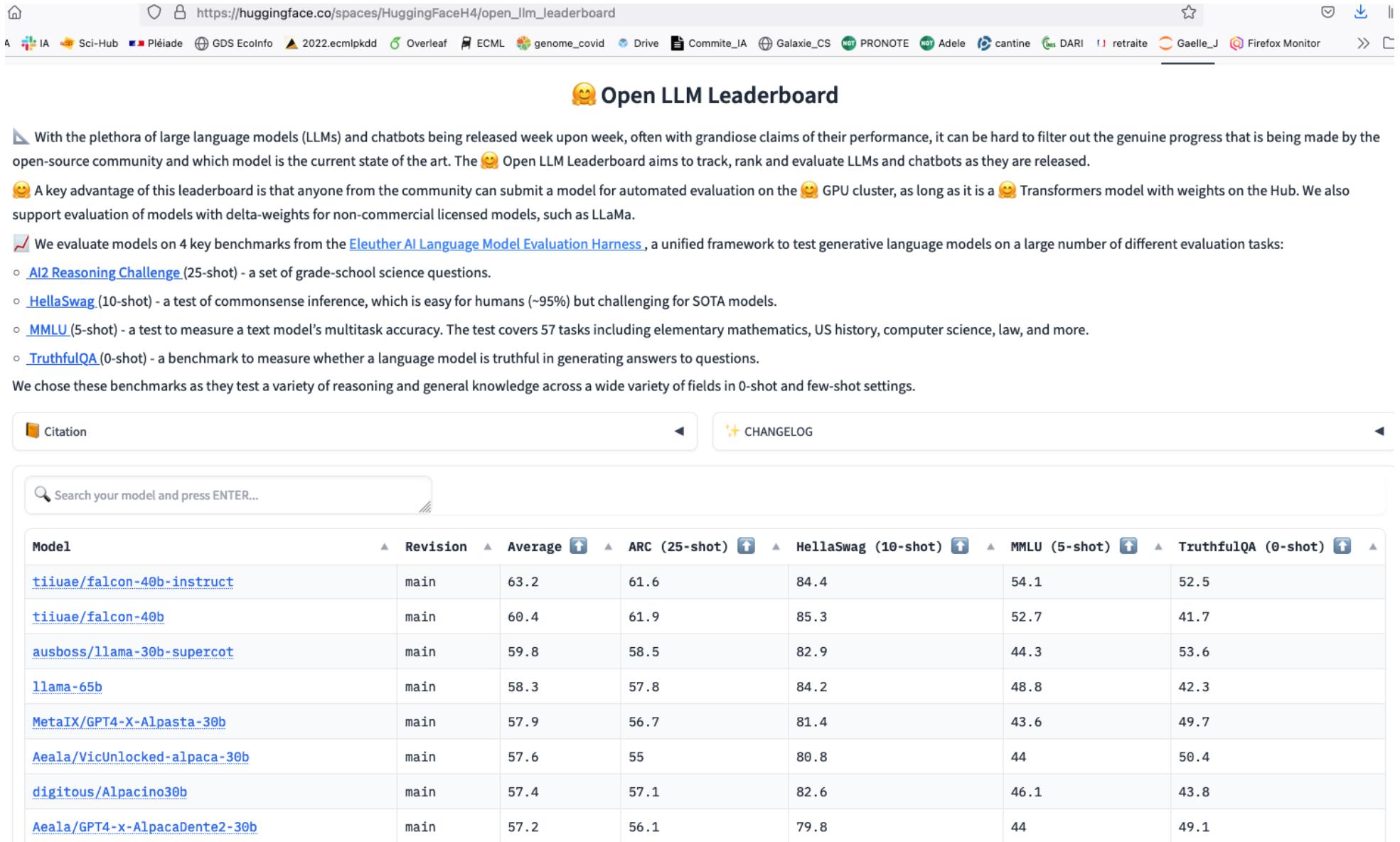


AI breakthroughs, last month...



...UAE Releases Open-Source **Falcon** 40B Generative AI LLM on May 25, 2023

And (for the moment) the winner is,



The screenshot shows a Firefox browser window displaying the "Open LLM Leaderboard" page at https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard. The page header features a smiling emoji and the title "Open LLM Leaderboard". Below the header, there is a section of text explaining the purpose of the leaderboard and its evaluation process. It mentions benchmarks like AI2 Reasoning Challenge, HellaSwag, MMLU, and TruthfulQA. A search bar is present above a table of model performance data.

With the plethora of large language models (LLMs) and chatbots being released week upon week, often with grandiose claims of their performance, it can be hard to filter out the genuine progress that is being made by the open-source community and which model is the current state of the art. The 😊 Open LLM Leaderboard aims to track, rank and evaluate LLMs and chatbots as they are released.

😊 A key advantage of this leaderboard is that anyone from the community can submit a model for automated evaluation on the 😊 GPU cluster, as long as it is a 😊 Transformers model with weights on the Hub. We also support evaluation of models with delta-weights for non-commercial licensed models, such as LLaMa.

✍ We evaluate models on 4 key benchmarks from the [EleutherAI Language Model Evaluation Harness](#), a unified framework to test generative language models on a large number of different evaluation tasks:

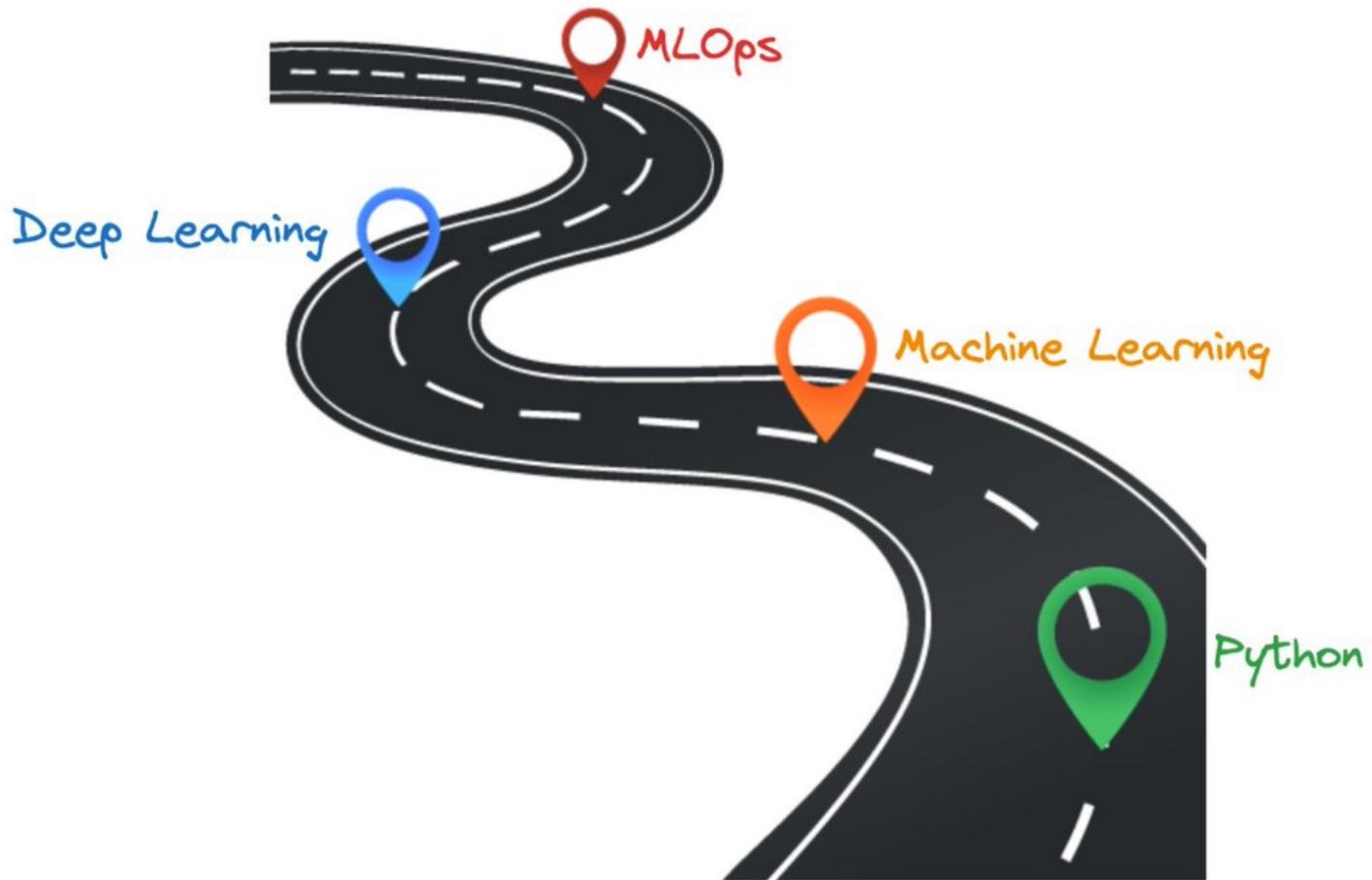
- [AI2 Reasoning Challenge](#) (25-shot) - a set of grade-school science questions.
- [HellaSwag](#) (10-shot) - a test of commonsense inference, which is easy for humans (~95%) but challenging for SOTA models.
- [MMLU](#) (5-shot) - a test to measure a text model's multitask accuracy. The test covers 57 tasks including elementary mathematics, US history, computer science, law, and more.
- [TruthfulQA](#) (0-shot) - a benchmark to measure whether a language model is truthful in generating answers to questions.

We chose these benchmarks as they test a variety of reasoning and general knowledge across a wide variety of fields in 0-shot and few-shot settings.

Model	Revision	Average	ARC (25-shot)	HellaSwag (10-shot)	MMLU (5-shot)	TruthfulQA (0-shot)
tiiuae/falcon-40b-instruct	main	63.2	61.6	84.4	54.1	52.5
tiiuae/falcon-40b	main	60.4	61.9	85.3	52.7	41.7
ausboss/llama-30b-supercot	main	59.8	58.5	82.9	44.3	53.6
llama-65b	main	58.3	57.8	84.2	48.8	42.3
MetaIX/GPT4-X-Alpasta-30b	main	57.9	56.7	81.4	43.6	49.7
Aeala/VicUnlocked-alpaca-30b	main	57.6	55	80.8	44	50.4
digitous/Alpacino30b	main	57.4	57.1	82.6	46.1	43.8
Aeala/GPT4-x-AlpacaDente2-30b	main	57.2	56.1	79.8	44	49.1

Introduction to AI: Road map

1. AI: a definition
2. AI and Machine learning
3. AI today
4. AI in cars
5. ChatGPT
6. Responsible AI



Autonomous vehicles: when?

According to Elon Musk: planned for 2014, 2015, 2016, ..., 2020, 2021, 2022, autonomous cars are now expected in 2023

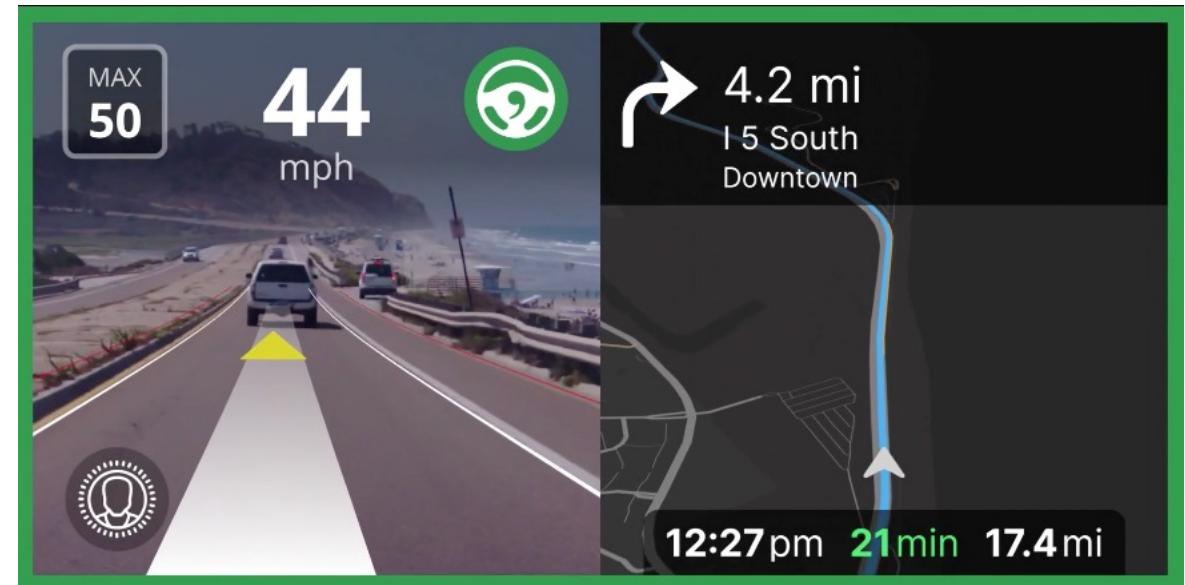


One day Elon Musk will be right...

ADAS AI components (e.g. OpenPilot)

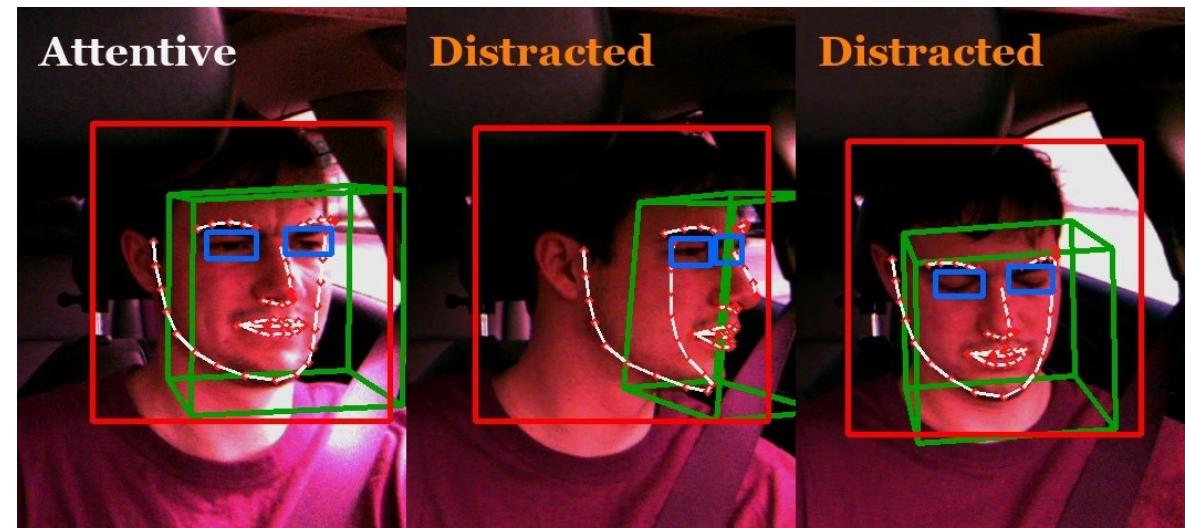
Diving agent

- Automated lane-centering
- Adaptive cruise control
- Assisted lane change
-



Driver monitoring system

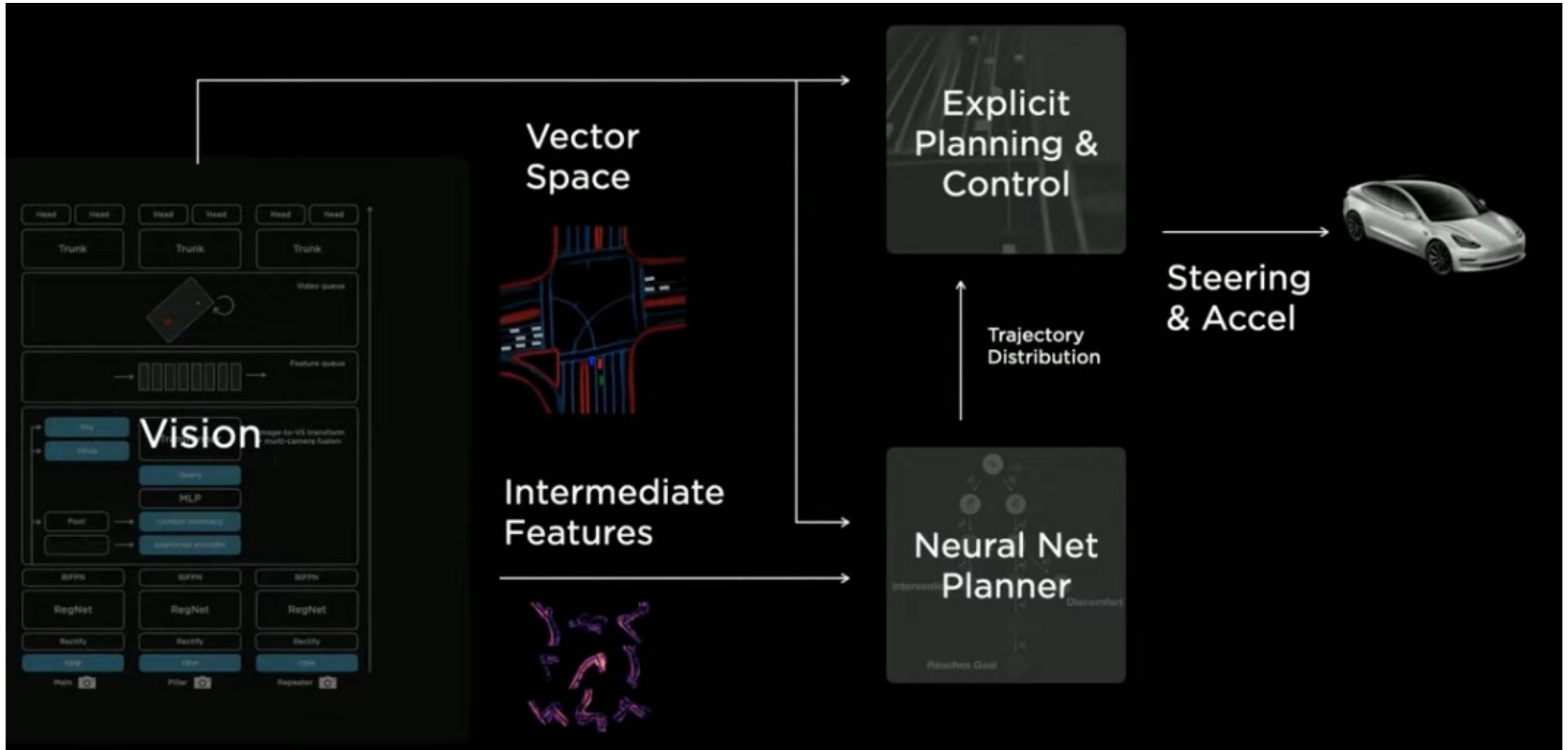
- Safety concerns



Anti Collision System

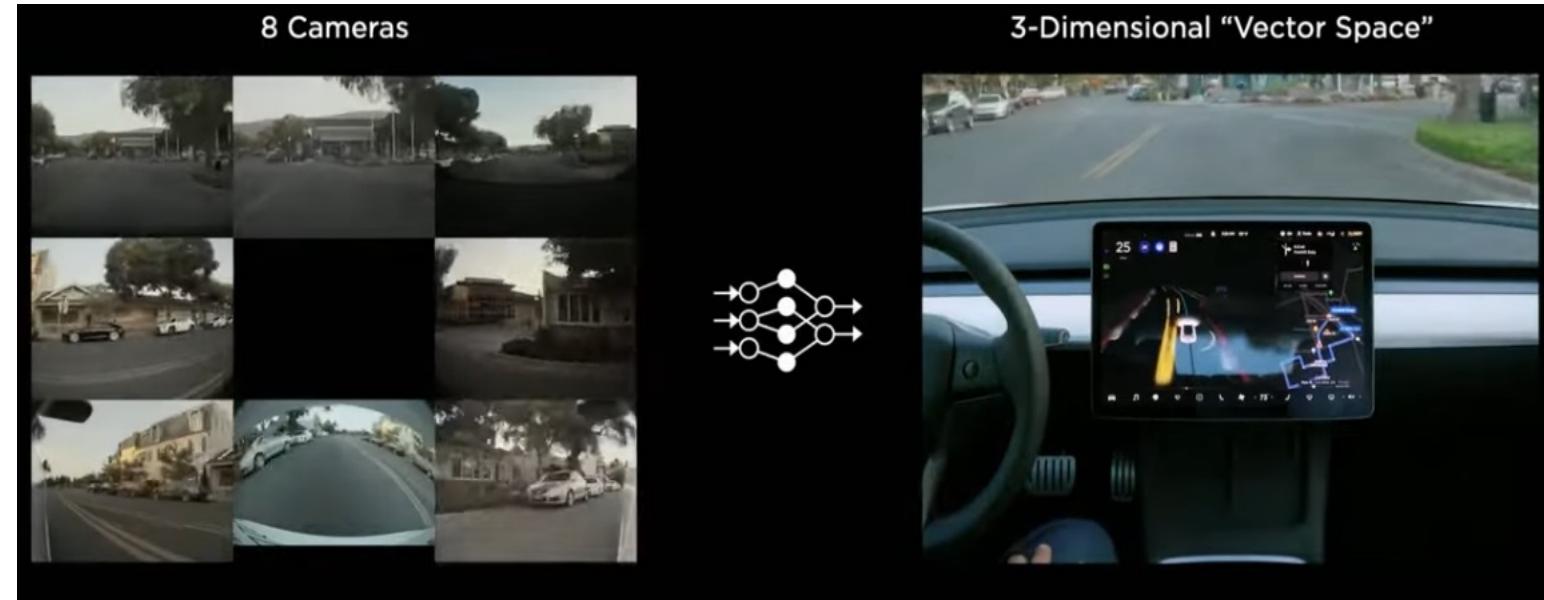
- Safety concerns

Tesla's driving agent



Tesla's perception module

1. Feature extractor: backbone



2. Multi scale feature fusion

3. Multi camera fusion

4. Time filtering

5. Multi task decision module

- item location (cars, pedestrian...)
- traffic signs (Stop sign, traffic light...)
- lane prediction
- ...

Tesla's perception module

1. Feature extractor: backbone
ResNet50 (2020), RegNet (2021)
(from a CVPR 2020 Facebook paper)
2. Multi scale feature fusion
EfficientDet
(from a 2019 Google paper)
3. Multi camera fusion
Transformers
(from a 2020 Facebook paper)
4. Time filtering
LSTM (recurrent neural network)
NIPS (1996)
5. Multi task decision module
 - item location (cars, pedestrian...)
 - traffic signs (Stop sign, traffic light...)
 - lane prediction
 - ...

This perception module contains: 48 networks, 1,000 outputs tensors, 70,000 GPU h to train
performs 40 prediction per second

An infinity of driving situations ...

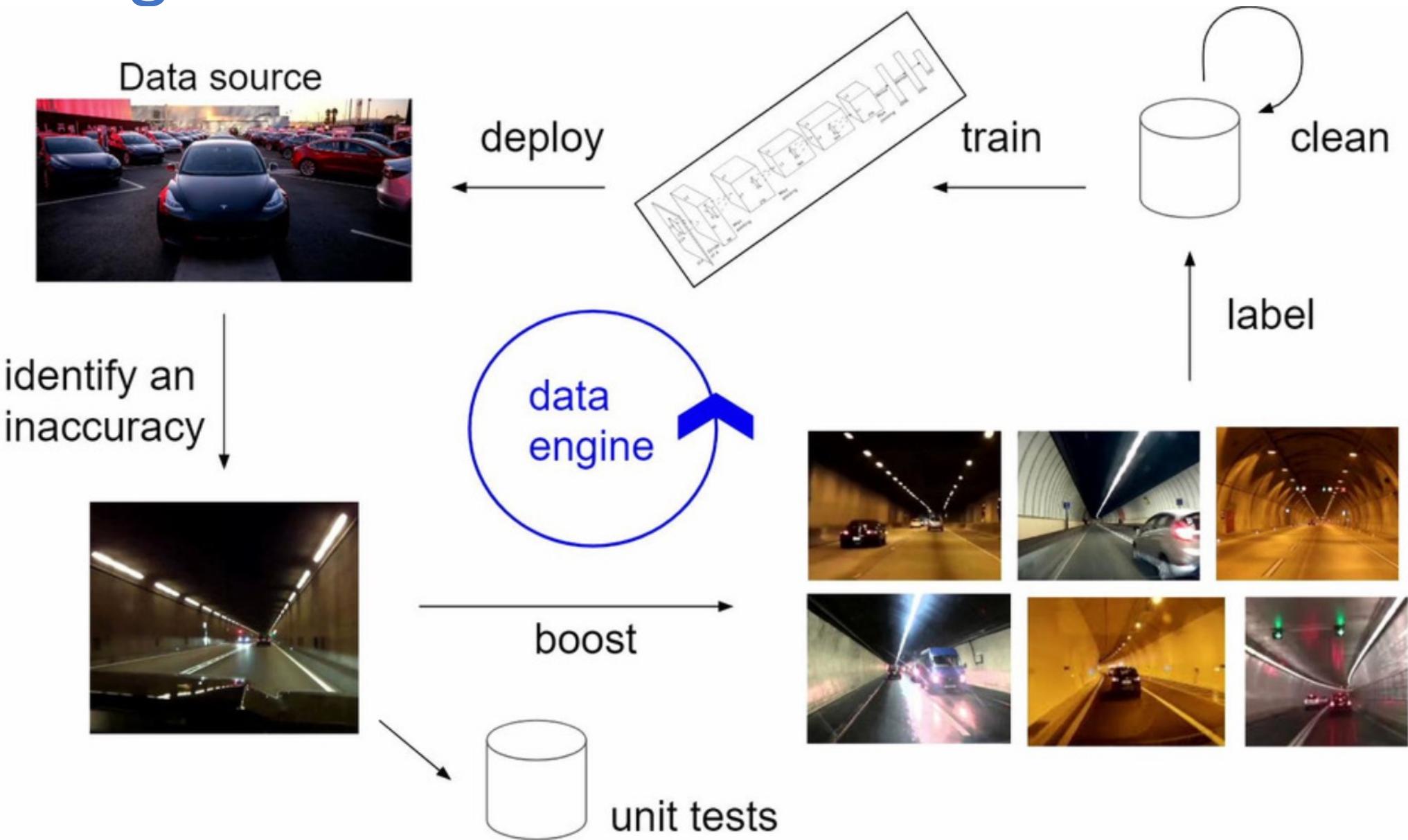


Taiwan, June 2020,



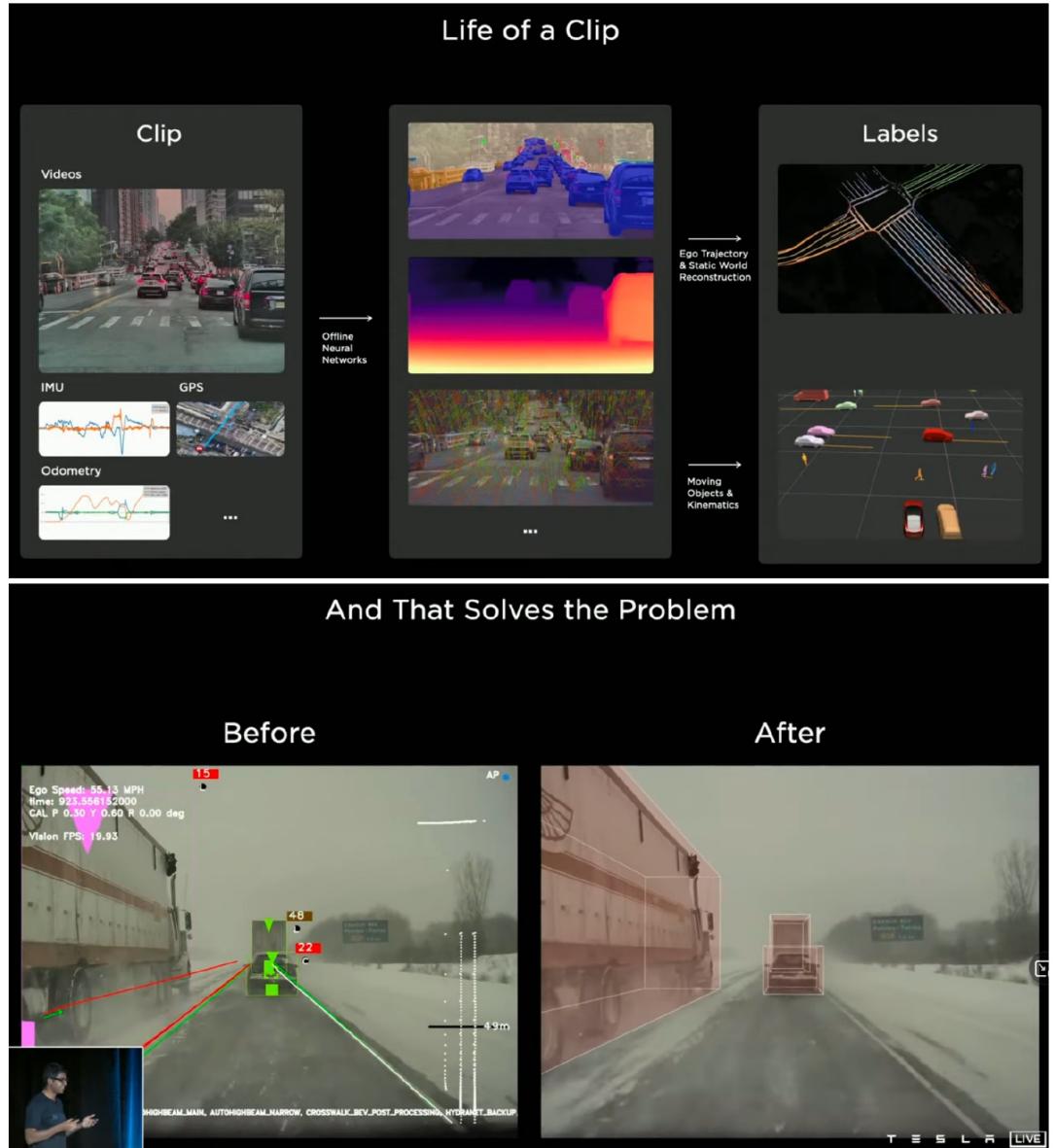
Andrej Karpathy - AI for Full-Self Driving at Tesla, Scaled ML, Feb 2020,

Data engine at Tesla



Data production at Tesla

- Gathering process
 - ▶ 221 triggering situations
- manual labelling (1000 person)
 - ▶ 2d -> 3d
 - ▶ reconstruction labelling
- auto labelling
 - ▶ use specifically trained networks
 - ▶ human to clean
- simulation
 - ▶ rare event
 - ▶ sensor robustness
 - ▶ adversarial examples



Tesla's FSD (hardware)



Model design and development
(at Tesla's headquarter)

Get more **data**

- data engine
- simulator
- auto labelling

Evaluate models

Training infra

- Dojo

Operational model
(in the car)

Driving agent

- perception
- planning

Driver monitoring

Inference infra

- FSD Chip (2019)

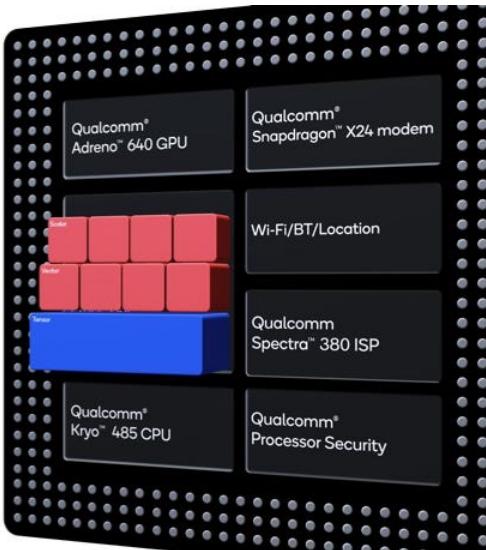
Tesla's Full self diving (beta): 35 releases, 281 models shipped, 75,000 models trained

Hardware: AI chips

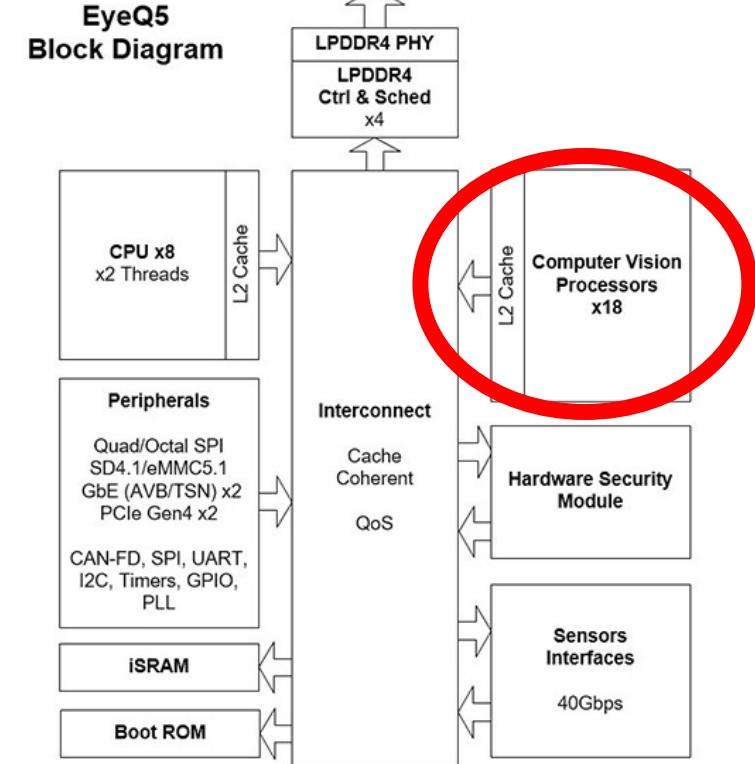
Apple
2022



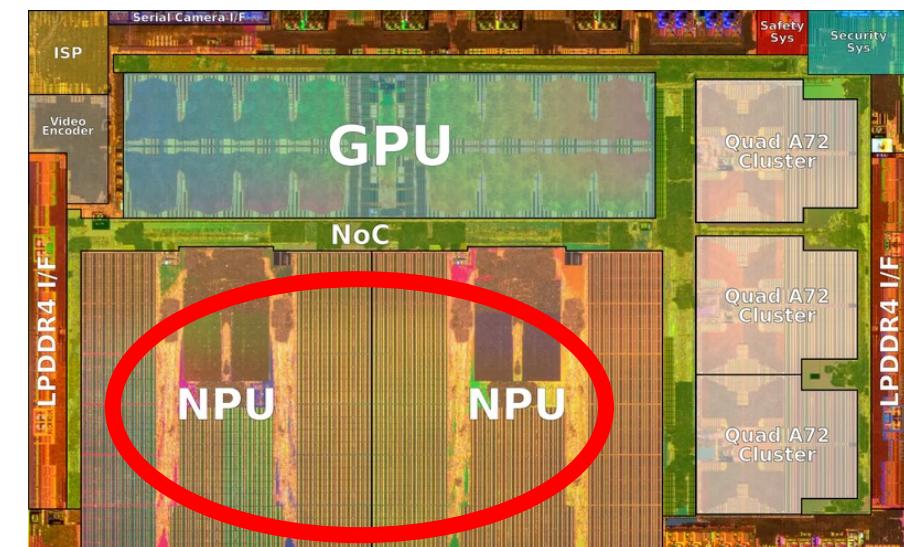
Qualcomm
2019



MobilEye
2018

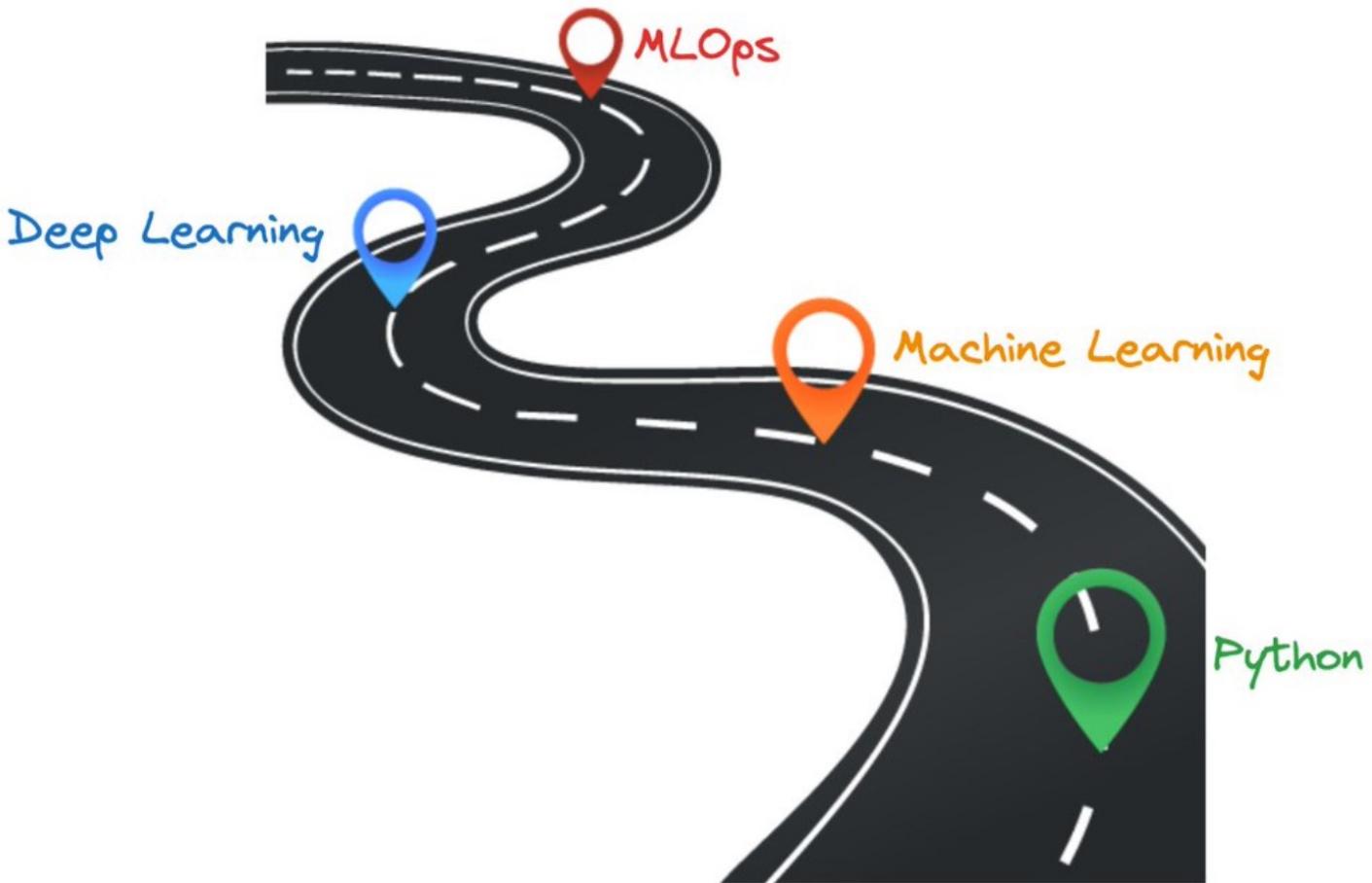


Tesla
2019



Introduction to AI: Road map

1. AI: a definition
2. AI and Machine learning
3. AI today
4. AI in cars
5. ChatGPT
6. Responsible AI



ChatGPT

SC

what is chatGPT?



ChatGPT is an advanced language model developed by OpenAI. It belongs to the GPT (Generative Pre-trained Transformer) family of models and is specifically designed for generating human-like text responses in a conversational manner. It has been trained on a massive amount of text data from various sources to develop a deep understanding of language patterns, grammar, context, and even some level of common sense reasoning.

ChatGPT can understand and generate text based on the given prompt or input. It can carry on coherent and contextually relevant conversations, answer questions, provide explanations, give suggestions, and engage in interactive dialogue with users. It can be utilized in a wide range of applications, such as customer support, language translation, content generation, and educational tools.

The model's responses are generated based on patterns and knowledge it has learned during training, but it's important to note that ChatGPT does not possess true understanding or consciousness. It generates responses purely based on statistical patterns in the training data and lacks real-time awareness or access to current information beyond its knowledge cutoff date.



Regenerate response

Send a message



Training ChatGPT

- 1) Generative pretraining
- 2) Supervised fine-tuning
- 3) Reinforcement learning from human feedback

1) Generative models

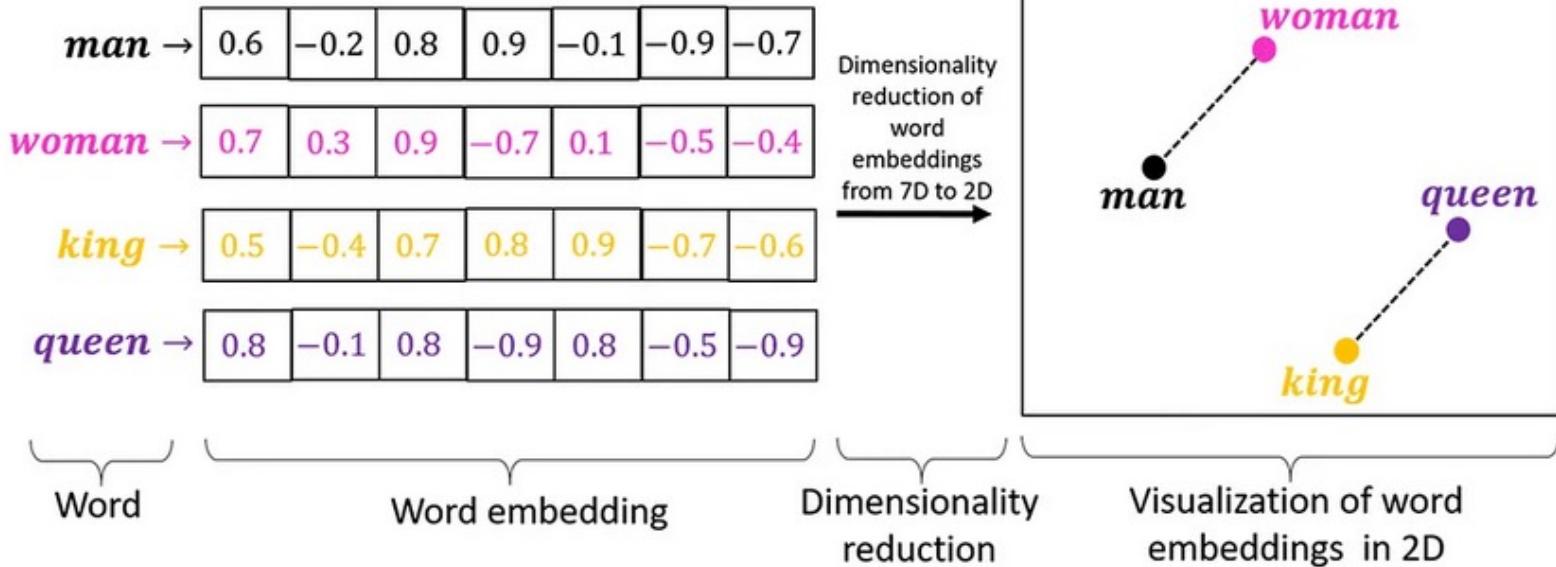
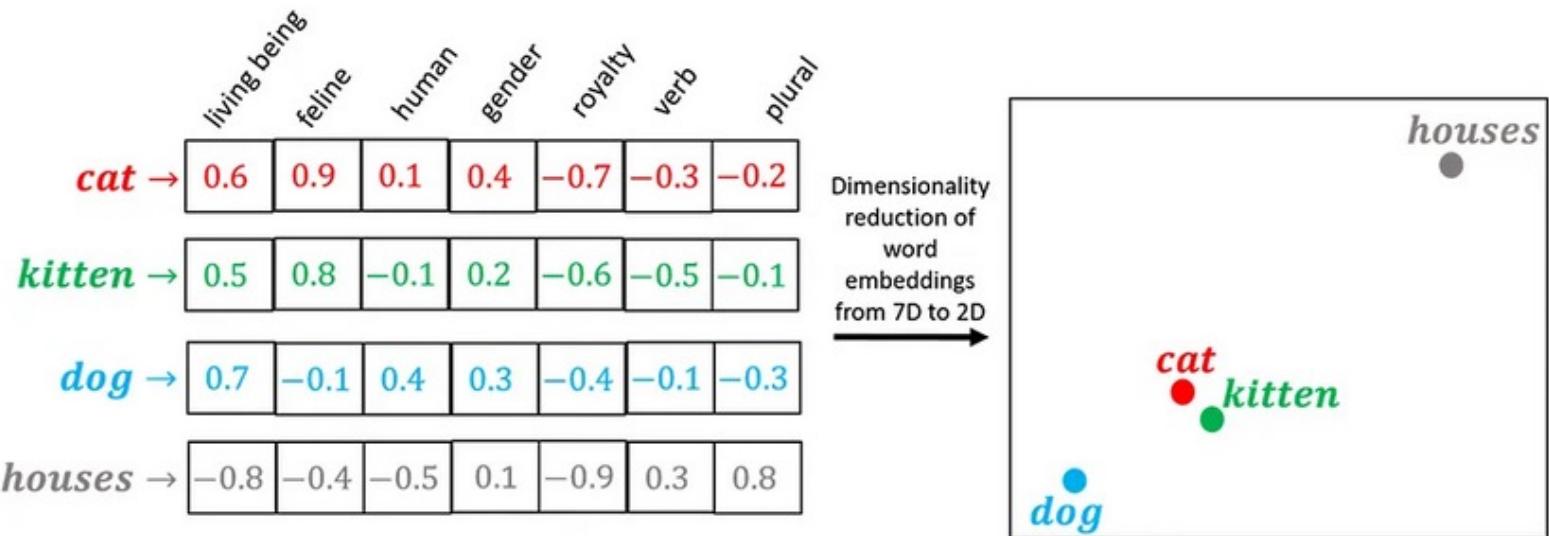
1.1 Represent

Word to vector (word2vec)

King – Man + Woman = Queen

1.2 Learn representations

Predict the next word

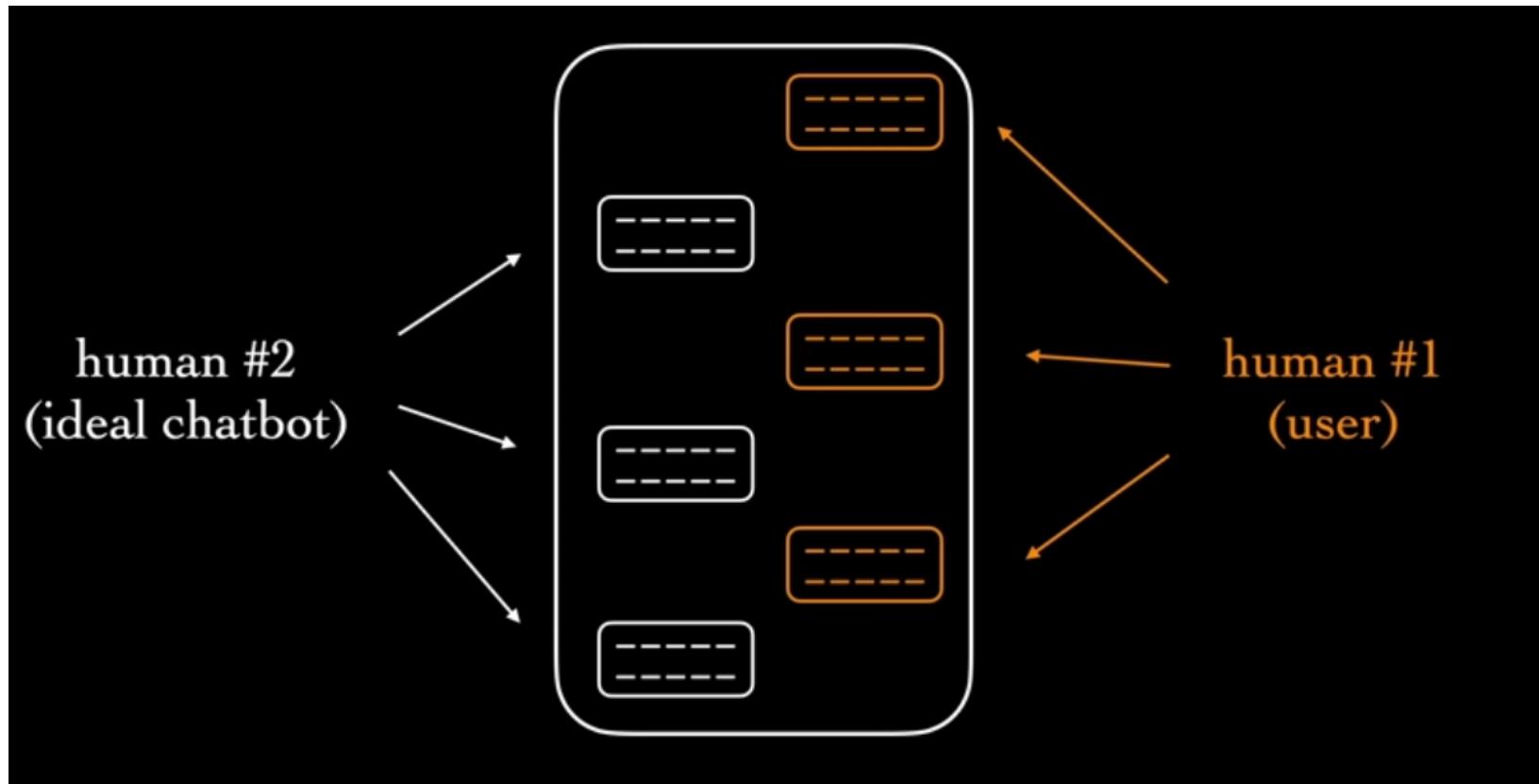


1) Generative model = foundation models

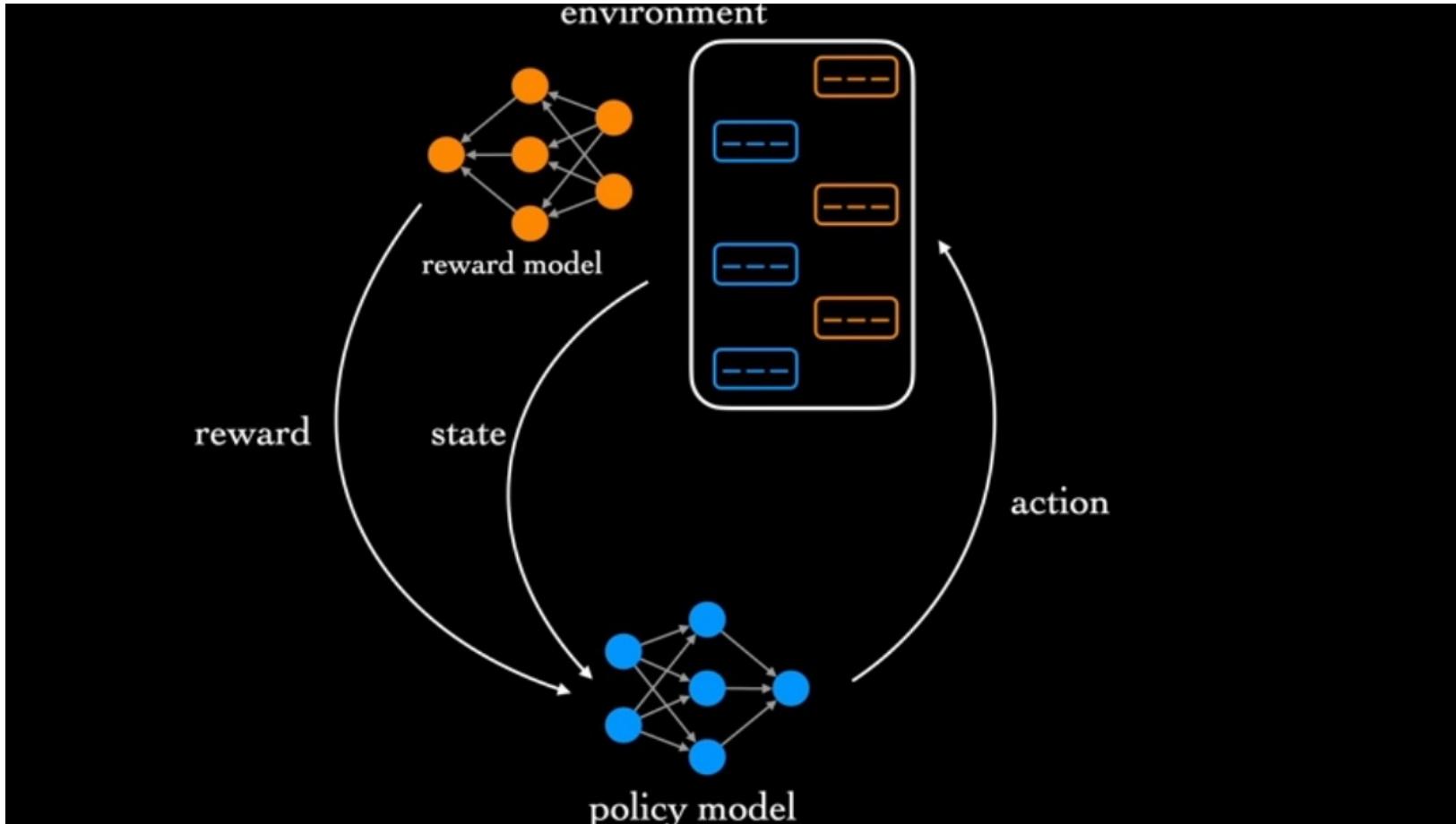
history (h):	next element:
Alice painted her house <u>?</u>	$P_{\theta}(\text{?} = \text{brown} \mid \textcolor{blue}{h}) = 0.2$ $P_{\theta}(\text{?} = \text{beige} \mid \textcolor{blue}{h}) = 0.1$ $P_{\theta}(\text{?} = \text{red} \mid \textcolor{blue}{h}) = 0.05$ $P_{\theta}(\text{?} = \text{because} \mid \textcolor{blue}{h}) = 0.09$ $P_{\theta}(\text{?} = \text{with} \mid \textcolor{blue}{h}) = 0.08$.

GPT: Generative pre-trained transformer (GPT 3 = 175 B parameters)
GPT 4 is multimodal (text+images)

2) Supervised Fine tuning

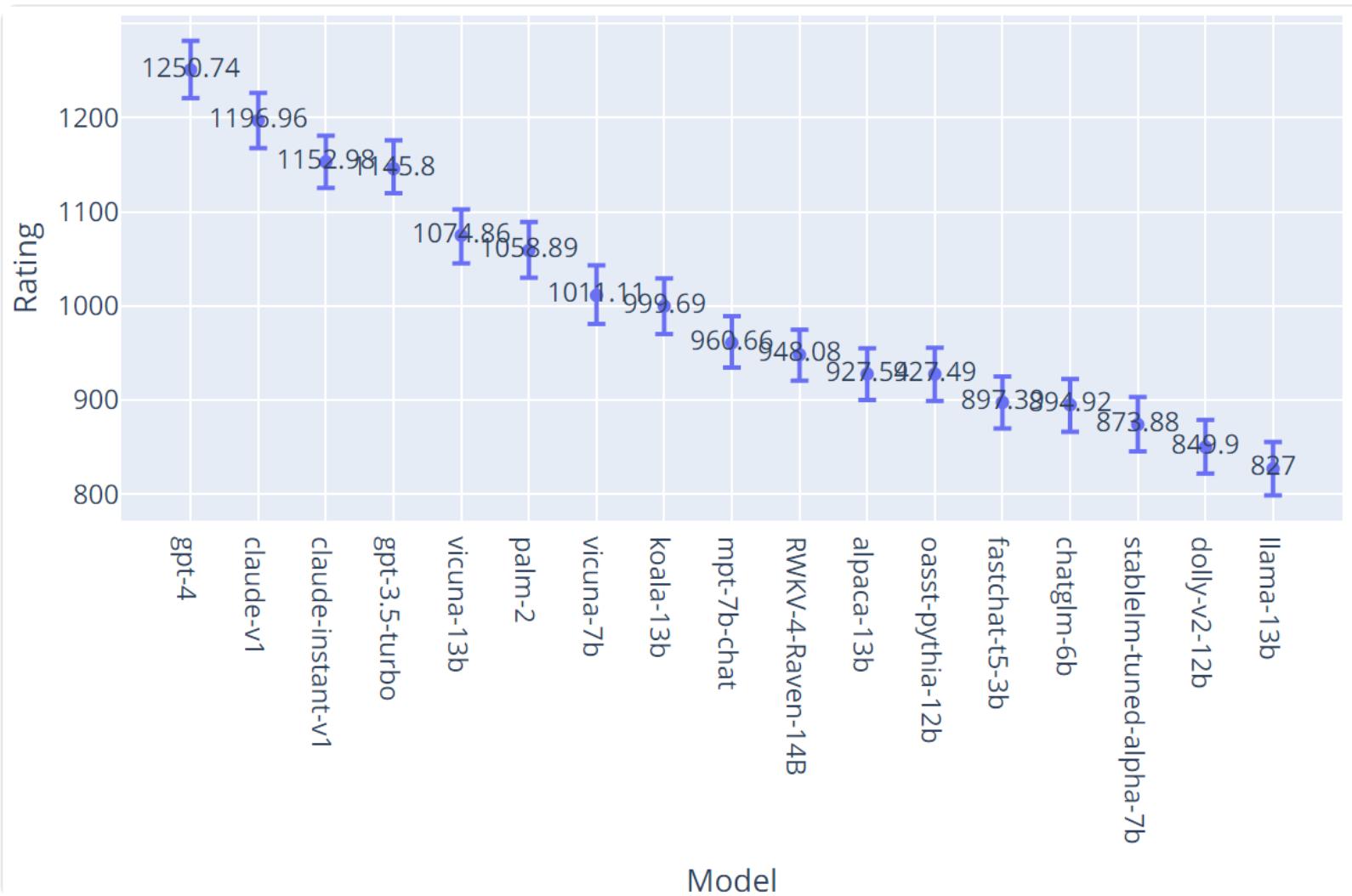


3) Reinforcement learning with human feedback Proximal Policy Optimization (PPO, Schulman, 2017)



What next...

Figure 4: Bootstrap of Elo Estimates (1000 Rounds of Random Sampling)



What next... try Faraday.dev

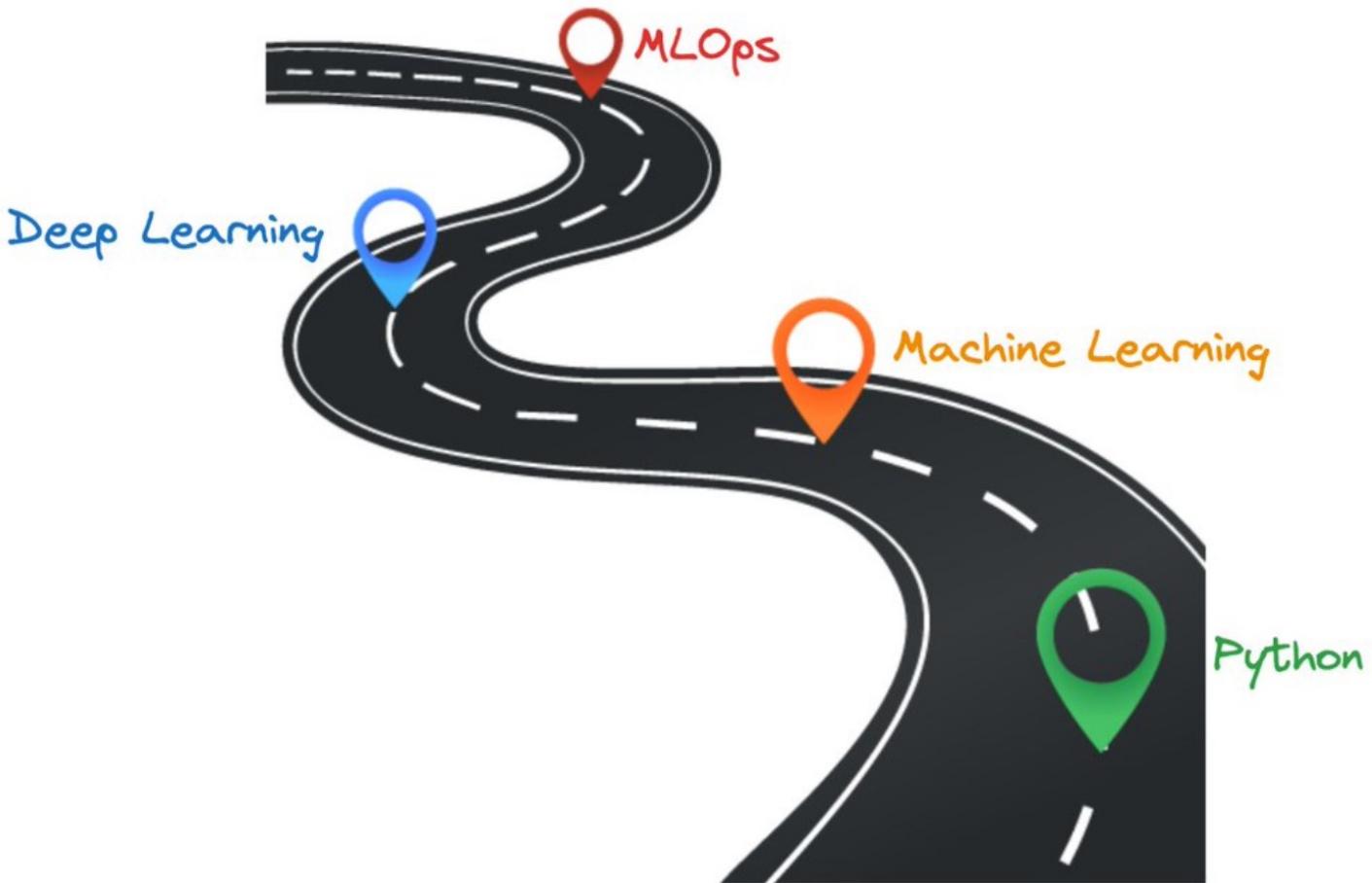
The screenshot shows the Faraday.dev AI chat interface. At the top, there are three colored dots (red, yellow, green) followed by the text "Faraday.dev". Next to it is a button labeled "Download Models". On the far right are links for "Discord" and "Twitter". Below this is a sidebar with a "+ New Chat" button and a list of previous messages from the Faraday AI. The messages include:

- what is Faraday.dev?
@ Faraday
- Untitled Chat
@ Faraday
- what is the size of
@ Faraday
- can you translate in
@ Faraday
- Untitled Chat
@ Faraday
- comment dit-on en an
@ Faraday
- do you speak french?
@ Faraday

The main chat area shows a message from the AI: "Hello User. How can I help you?". A user message "what is Faraday.dev?" is shown in a dashed box. The AI's response is: "faraday.dev is an open source platform that allows developers to create powerful web applications using their preferred programming language. It provides users with a simple interface to deploy and manage their projects." At the bottom, there are "Undo" and "Regenerate" buttons, and a text input field with placeholder text "Type your message here...". A small note at the bottom states: "Model may produce inaccurate information about people, places, or facts. For research purposes only." and "Model: WizardLM 1.0 13B (Q5_1)".

Introduction to AI: Road map

1. AI: a definition
2. AI and Machine learning
3. AI today
4. AI in cars
5. ChatGPT
6. **Responsible AI**



Environmental impact of AI

Hard to measure:

First (usage), second (applicative) and third (systemic) order impact

Usage:

device, datacenter, network + life cycle (transportation, recycling...)

=> material, manufacturing + energy

Case studies

NLP [Strubell et al., 2019]

- 4 state-of-the-art models in automatic language processing (NLP)
- Software-based energy cost estimation
- emissions: between 18kg eqCO₂ and 284,000 kg eqCO₂
- most used model then: 652 kg eqCO₂, (one way 2500 km by car)

Bloom (open source generative model)

- Training = 4 months on Jean ZAY (GPU = 4M€)
- Jean Zay is energetically efficient, cost = 400 k€

Case studies

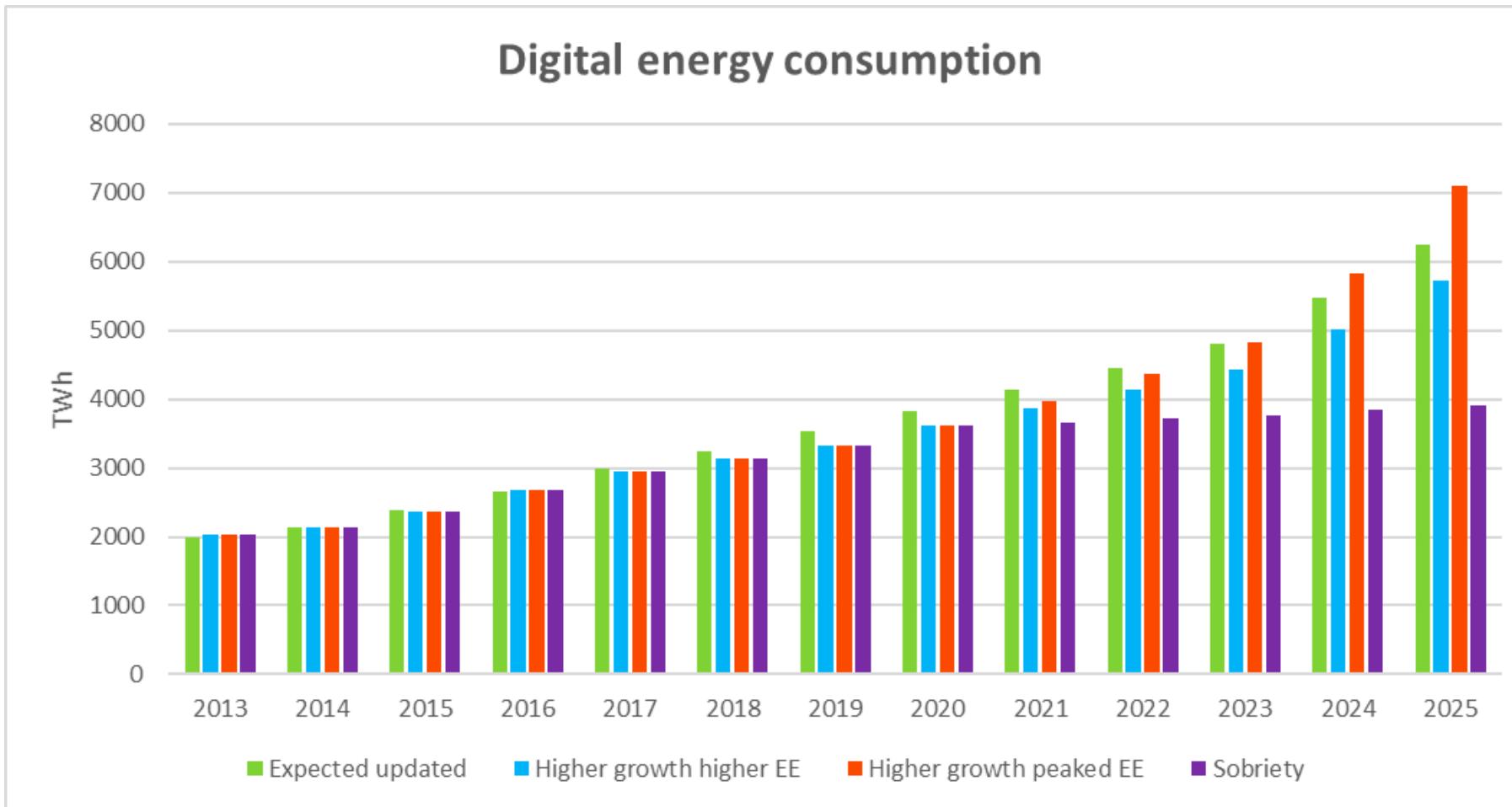
NLP [Strubell et al., 2019]

- 4 state-of-the-art models in automatic language processing (NLP)
- Software-based energy cost estimation
- emissions: between 18kg eqCO₂ and 284,000 kg eqCO₂
- most used model then: 652 kg eqCO₂, (one way 2500 km by car)

Bloom (open source generative model)

- Training = 4 months on Jean ZAY (GPU = 4M€)
- Jean Zay is energetically efficient, cost = 400 k€

Towards reasonable sobriety



The Shift Project from data published by (Andrae & Edler, 2015)]

Montréal Declaration for a Responsible Development of IA

[Dilhac et al., 2018a]

10- SUSTAINABLE DEVELOPMENT PRINCIPLE

The development and use of AIS must be carried out so as to ensure a strong environmental sustainability of the planet.

- 1) AIS hardware, its digital infrastructure and the relevant objects on which it relies such as data centers, must aim for the greatest energy efficiency and to mitigate greenhouse gas emissions over its entire life cycle.
- 2) AIS hardware, its digital infrastructure and the relevant objects on which it relies, must aim to generate the least amount of electric and electronic waste and to provide for maintenance, repair, and recycling procedures according to the principles of circular economy.
- 3) AIS hardware, its digital infrastructure and the relevant objects on which it relies, must minimize our impact on ecosystems and biodiversity at every stage of its life cycle, notably with respect to the extraction of resources and the ultimate disposal of the equipment when it has reached the end of its useful life.
- 4) Public and private actors must support the environmentally responsible development of AIS in order to combat the waste of natural resources and produced goods, build sustainable supply chains and trade, and reduce global pollution.

Red vs Green AI

[Schwartz et al., 2019]

The equation of **Red AI**: The cost of an AI (R) result grows linearly with the cost of processing a single (E)xample, the size of the training (D)ataset and the number of (H)yperparameter experiments.

$$\text{Cost}(R) = E \ D \ H$$

The diagram illustrates the components of the Red AI cost equation. At the top, the equation $\text{Cost}(R) = E \ D \ H$ is displayed. Below it, three labels are positioned under the variables: "network" under E , "data set" under D , and "hyperparameters" under H . Red arrows point from each label to its corresponding variable in the equation.

Green AI Measures Efficiency

small and sober

fine tune

pre trained

Sober neural networks

Small architecture

- Pruning/sparse architecture
- Model compression (distillation)
- Pre training/few shot learning

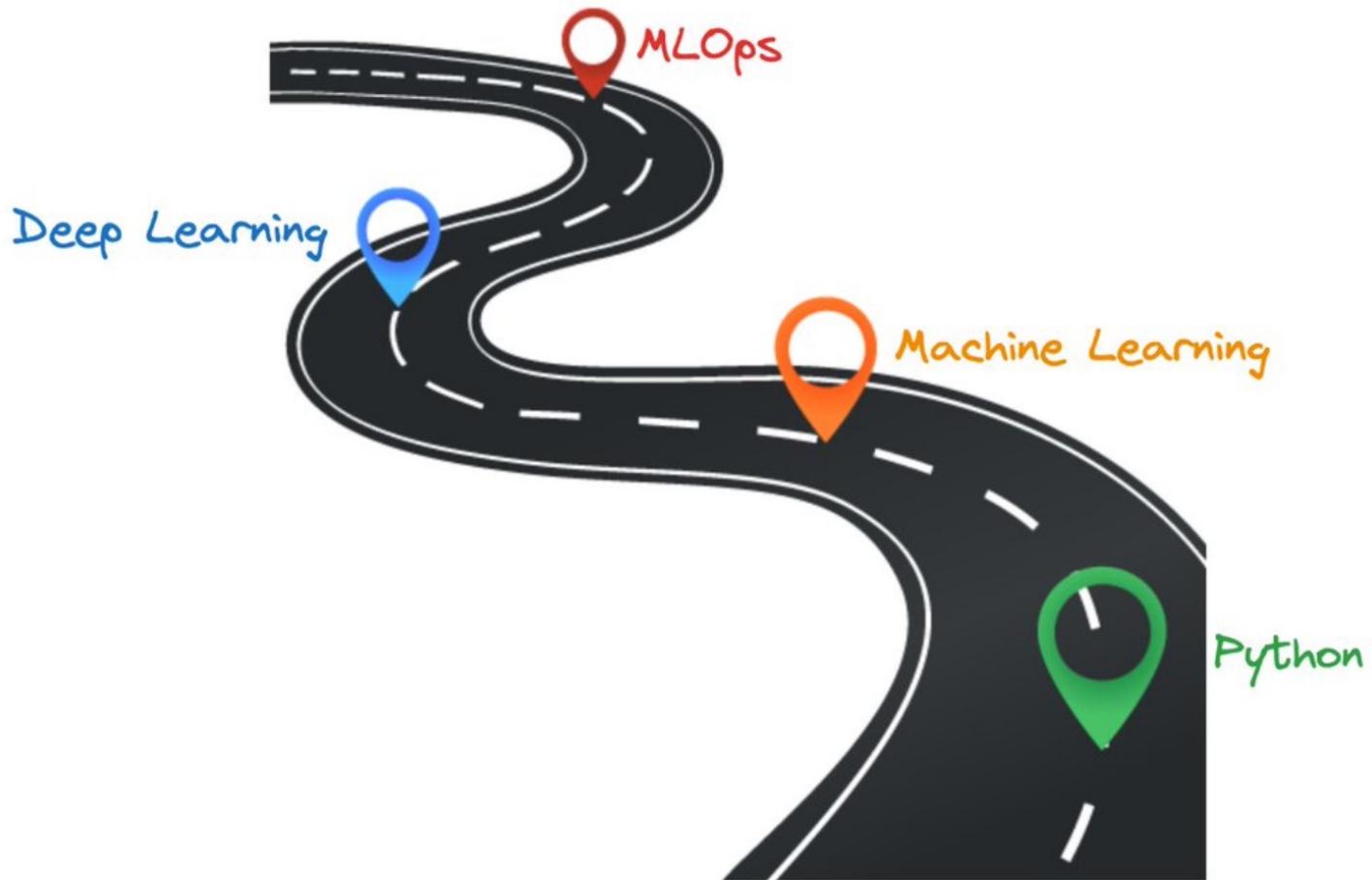
Implementation

- Quantization (Float AI)
- Algorithm and Hardware Codesign (NPU)

Embedded IA vs cloud computing

Introduction to AI: Road map done

1. AI: a definition
2. AI and Machine learning
3. AI today
4. AI in cars
5. ChatGPT
6. Responsible AI



Conclusion

- Today's AI is Deep Learning

As Michael Jordan (2018) ironically stated, “in an interesting reversal, it is Wiener’s intellectual agenda that has come to dominate in the current era, under the banner of McCarthy’s terminology”.

- Today's AI is open and improving every day
- AI is far from replacing human intelligence, but is may enhance certain skills
- Breakthrough are expected, ...but when?

The screenshot shows the GitHub Copilot interface integrated into a Visual Studio Code (VS Code) workspace. The left sidebar displays the GitHub Copilot welcome message, which includes instructions for asking questions, generating unit tests, explaining selected code, and proposing fixes for bugs. The main editor area shows a Python script named `add_elements.py`. The script defines a function `parse_expenses` that takes a string of expense data and returns a list of tuples. The function uses a for loop to iterate over lines, ignoring lines starting with '#'. It then splits each line by space to extract date, value, and currency, converts the date to a datetime object, and appends the tuple to the expenses list. A sample data string is provided at the bottom. The status bar at the bottom right indicates that the file is in Python mode.

```
def parse_expenses(expenses_string):  
    """Parse the list of expenses and return the list of triples  
    (date, value, currency).  
    Ignore lines starting with #.  
    Parse the date using datetime.  
    Example expenses_string:  
        2023-01-02 -34.01 USD  
        2023-01-03 2.59 DKK  
        2023-01-03 -2.72 EUR  
    """  
  
    expenses = []  
    for line in expenses_string.splitlines():  
        if line.startswith("#"):  
            continue  
        date, value, currency = line.split(" ")  
        expenses.append((float(value),  
                        currency,  
                        datetime.datetime.strptime(date,  
                            "%Y-%m-%d")))  
  
    return expenses  
  
expenses_data = '''2023-01-02 -34.01 USD  
2023-01-03 2.59 DKK  
2023-01-03 -2.72 EUR'''
```