Gen AI & Classic ML

when to use what

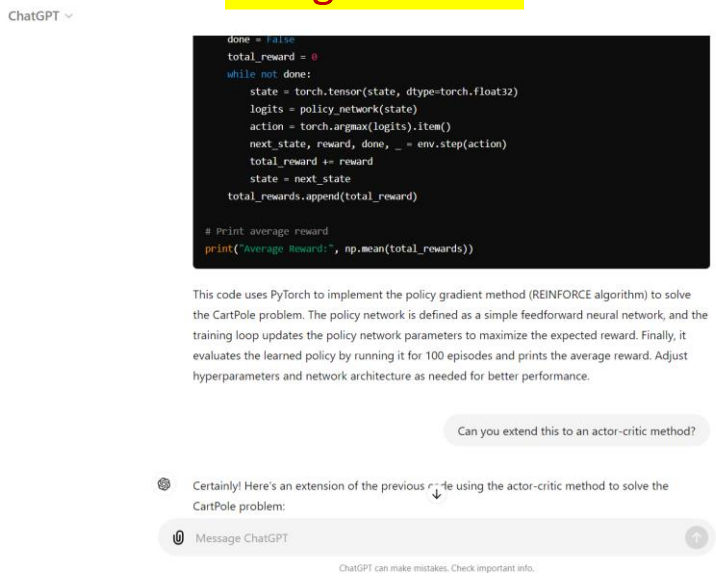
October 2024

Felix Wick

Classic ML vs Deep Learning

Generative vs Predictive Models

text generation



[ChatGPT](#)

image synthesis

Prompt: Epic anime artwork of a wizard atop a mountain at night casting a cosmic spell into the dark sky that says "Stable Diffusion 3" made out of colorful energy

[Stable Diffusion 3 — Stability AI](#)

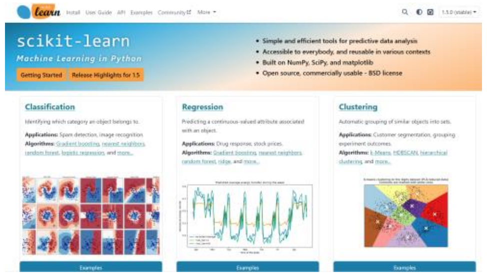
text-to-video

Prompt: A stylish woman walks down a Tokyo street filled with warm glowing neon and animated city signage. She wears a black leather jacket, a long red dress, and black boots, and carries a black purse. She wears sunglasses and red lipstick. She...

[Sora | OpenAI](#)

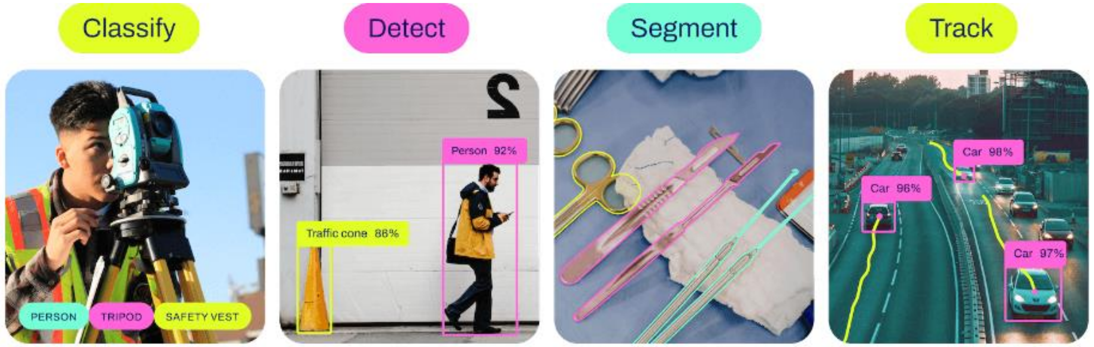
BERT family

tabular data



dmlc XGBoost

computer vision



[YOLO](#)

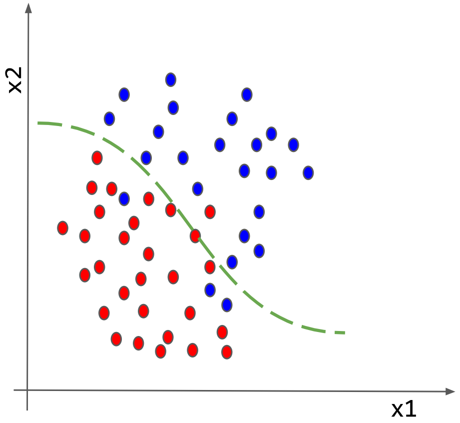
Deep Learning & Foundation Models

Classic Supervised Learning

input-output mapping

specific models for each task and data set

feature engineering



Example: Spam Filtering

Classify emails as spam or no spam

*use accordingly **labeled emails as training set***

use information like **occurrence of specific words** or **email length** as **features**

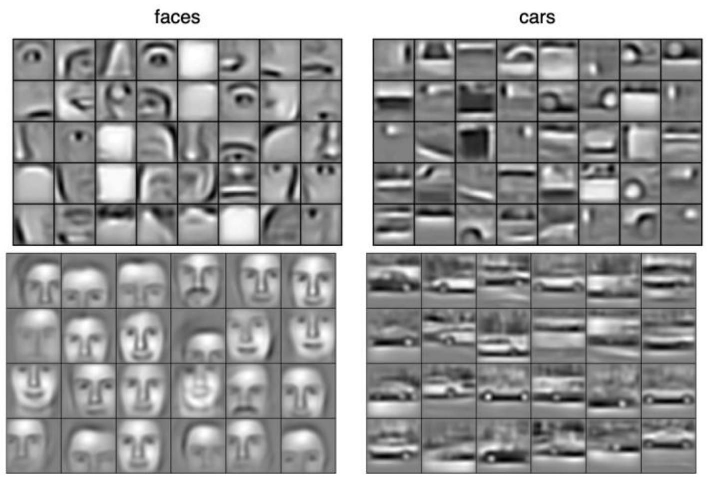
features x1 and x2
spam, no spam

Ladder of Generalization

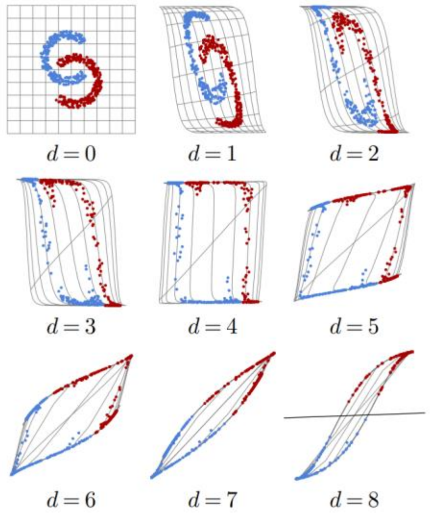
classic ML: feature engineering

deep learning: feature learning

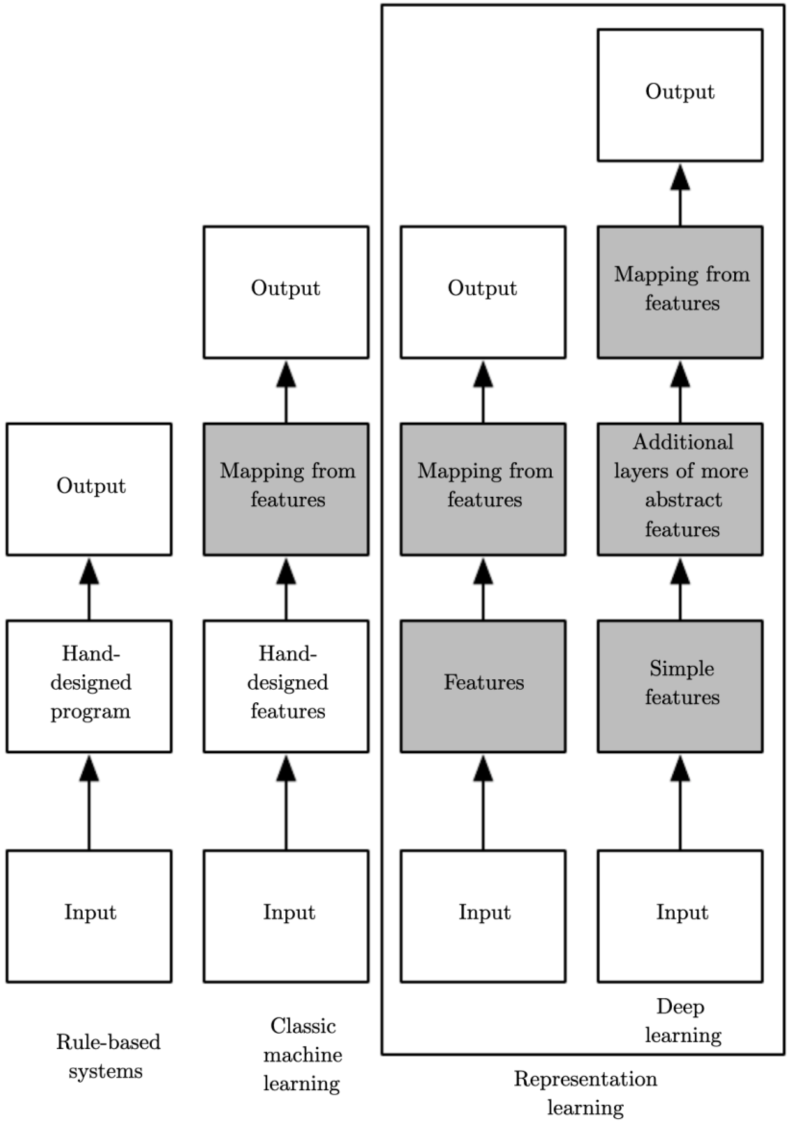
(hierarchy of concepts learned from raw data in deep graph with many layers)



source



source



source

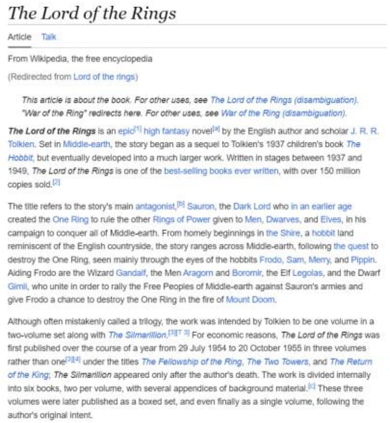
Structured/Tabular vs Unstructured Data

unstructured data: homogenous

→ deep learning rules

→ allows transfer learning (foundation models in CV and NLP)

ImageNet



The Lord of the Rings

Article Talk

From Wikipedia, the free encyclopedia

(Redirected from Lord of the rings)

This article is about the book. For other uses, see The Lord of the Rings (disambiguation).
"War of the Ring" redirects here. For other uses, see War of the Ring (disambiguation).

The Lord of the Rings is an epic[1] high fantasy novel[a] by the English author and scholar J. R. R. Tolkien. Set in Middle-earth, the story began as a sequel to Tolkien's 1937 children's book *The Hobbit*, but eventually developed into a much larger work. Written in stages between 1937 and 1949, *The Lord of the Rings* is one of the best-selling books ever written, with over 150 million copies sold.[2]

The title refers to the story's main antagonist,[b] Sauron, the Dark Lord who in an earlier age created the One Ring to rule the other Rings of Power given to Men, Dwarves, and Elves, in his campaign to conquer all of Middle-earth. From homely beginnings in the Shire, a hobbit land reminiscent of the English countryside, the story ranges across Middle-earth, following the quest to destroy the One Ring, seen mainly through the eyes of the hobbits Frodo, Sam, Merry, and Pippin. Aiding Frodo are the Wizard Gandalf, the Men Aragorn and Boromir, the Elf Legolas, and the Dwarf Gimli, who unite in order to rally the Free Peoples of Middle-earth against Sauron's armies and give Frodo a chance to destroy the One Ring in the fire of Mount Doom.

Although often mistakenly called a trilogy, the work was intended by Tolkien to be one volume in a two-volume set along with *The Silmarillion*.[3][T 3] For economic reasons, *The Lord of the Rings* was first published over the course of a year from 29 July 1954 to 20 October 1955 in three volumes rather than one[3][4] under the titles *The Fellowship of the Ring*, *The Two Towers*, and *The Return of the King*; *The Silmarillion* appeared only after the author's death. The work is divided internally into six books, two per volume, with several appendices of background material.[c] These three volumes were later published as a boxed set, and even finally as a single volume, following the author's original intent.

structured data: heterogenous

→ feature engineering needed

→ deep learning loses its advantage over shallow methods

→ e.g., gradient boosting still used a lot

	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	...	PoolArea	PoolQC	Fence	MiscFeature	MiscVal	MoSold	YrSold	SaleType	SaleCondition	SalePrice
0	1	60	RL	65.0	8450	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN	NaN	0	2	2008	WD	Normal	208500
1	2	20	RL	80.0	9600	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN	NaN	0	5	2007	WD	Normal	181500
2	3	60	RL	68.0	11250	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	9	2008	WD	Normal	223500
3	4	70	RL	60.0	9550	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	2	2006	WD	Abnorml	140000
4	5	60	RL	84.0	14260	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	12	2008	WD	Normal	250000
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1455	1456	60	RL	62.0	7917	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN	NaN	0	8	2007	WD	Normal	175000
1456	1457	20	RL	85.0	13175	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	MnPrv	NaN	0	2	2010	WD	Normal	210000
1457	1458	70	RL	66.0	9042	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	GdPrv	Shed	2500	5	2010	WD	Normal	266500
1458	1459	20	RL	68.0	9717	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN	NaN	0	4	2010	WD	Normal	142125
1459	1460	20	RL	75.0	9937	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN	NaN	0	6	2008	WD	Normal	147500

[1460 rows x 81 columns]

Transfer Learning

idea:
- generic pre-training of foundation models on huge data sets
- subsequent finetuning for specific tasks on small(er) data sets

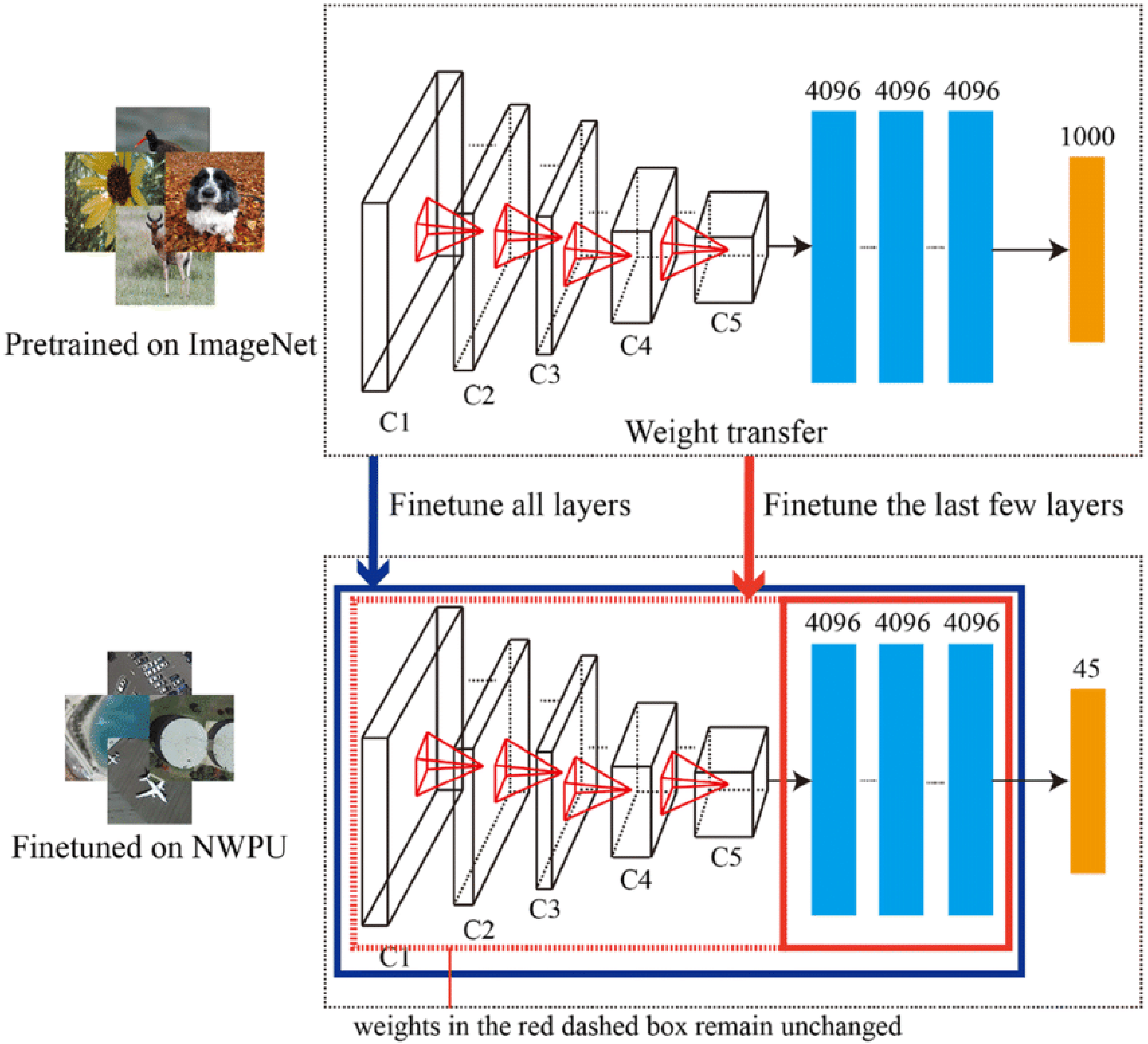
(usually done with deep learning methods, using its compositional nature)

very successful for:
- computer vision (e.g., object classification)
- language models (e.g., BERT, GPT)

not (yet) for tabular data (due to its notorious heterogeneity)

(but some promising research, e.g., Chronos for univariate time series)

CNN Finetuning



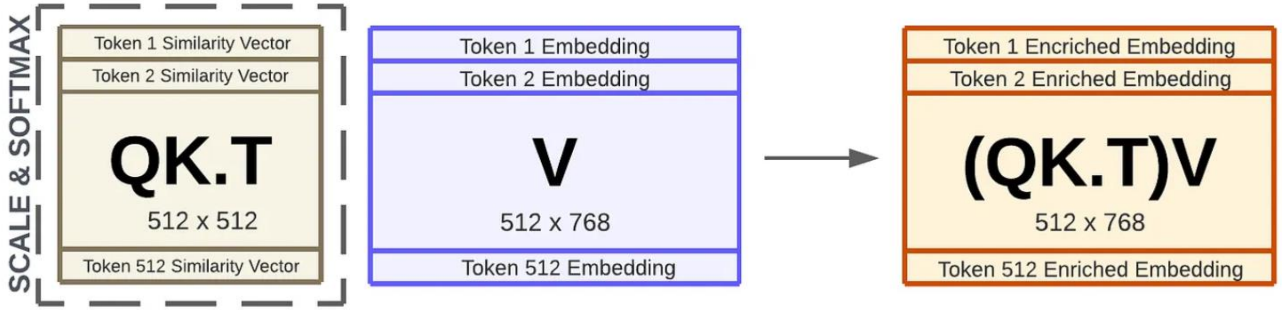
[source]

These are no generative, but predictive models.

But aspects of it can be applied in generative ones (e.g., U-Net in diffusion models).

Language Models: Contextual Semantics

- self-supervised learning: e.g., next/masked-word prediction
- tokenization: split text into chunks (e.g., words)
- semantics by means of vector embeddings: e.g., via bag-of-words (or end-to-end in transformer)
- positional encoding & embeddings: order of sequence
- contextual embeddings: (self-)attention (weighted averages: influence from other tokens)



source

most new LLMs decoder-only

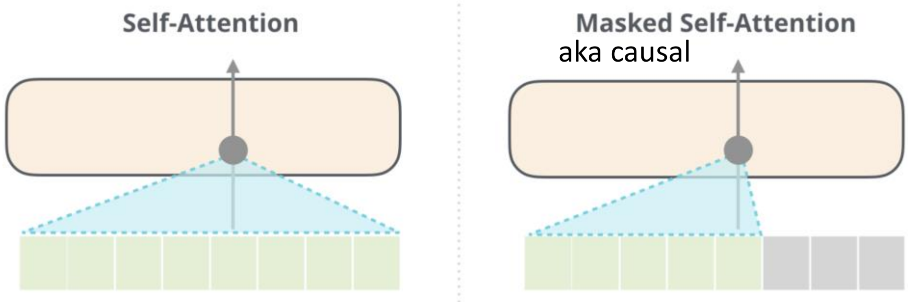
Encoder vs Decoder LLMs

encoder-only LLMs

- prime example: BERT
- self-supervised pre-training: masked-word prediction
- finetuning on downstream tasks (e.g., sequence classification)
- can't generate text
- can't be prompted

decoder-only LLMs

- prime example: GPT
- self-supervised pre-training: next-word prediction
- instruction tuning (e.g., RL from human feedback)
- generate text: chat bots
- prompt engineering (zero-/few-shot)



source

Generative AI

Generative vs Predictive/Discriminative Models

discriminative models:

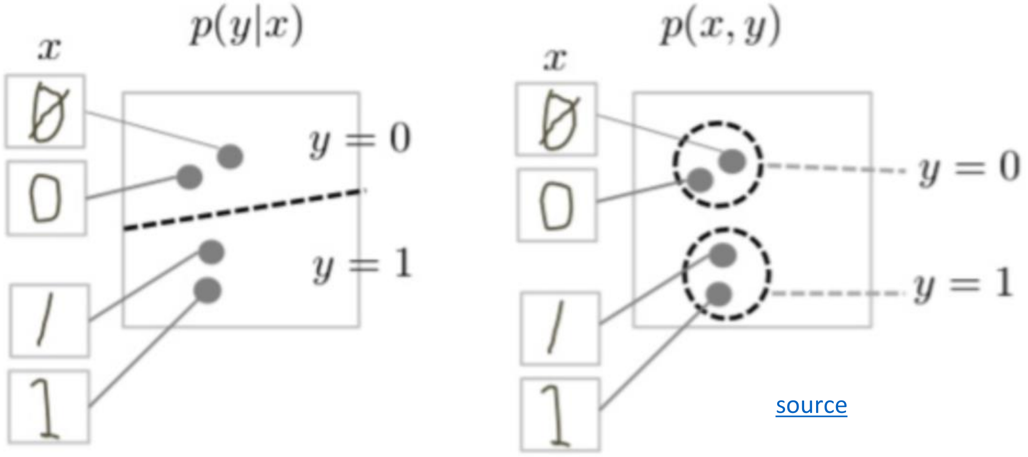
predict conditional probability $P(Y|\boldsymbol{X})$

generative models:

predict joint probability $P(Y, \boldsymbol{X})$

(or just $P(\boldsymbol{X})$ → unsupervised learning)

discriminative model generative model



source

task of generative models more difficult: need to model full data distribution rather than merely find patterns in inputs to distinguish outputs

generative models allow to generate new data samples (text, images, video, proteins, …)

predictive models usually better for predictive tasks, business problems often specific/predictive

Deep Learning for Generative AI

Depending on the application, there are currently two dominant approaches for generative AI:

- text generation: decoder LLMs

- image synthesis: diffusion models

note the difference between image synthesis and multimodal understanding in LLMs (images as additional input sequences to transformer, tokenized by splitting into patches)

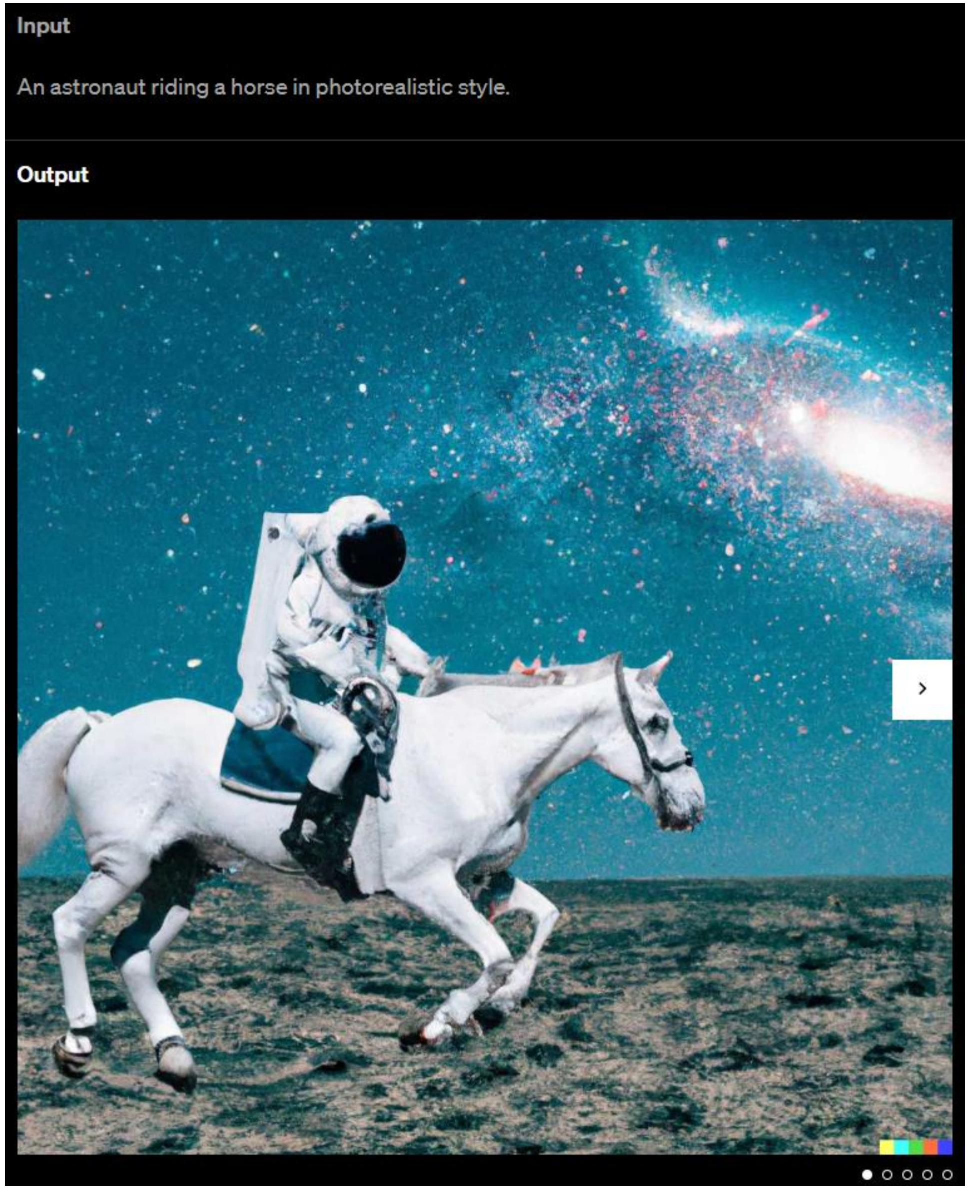
Image Synthesis

idea: generate new images as variations of training data

usually conditioned on text (prompt) by transformers

compared to text generation, additional mechanism needed (e.g., diffusion) to create more complex structures

example: DALL-E 2

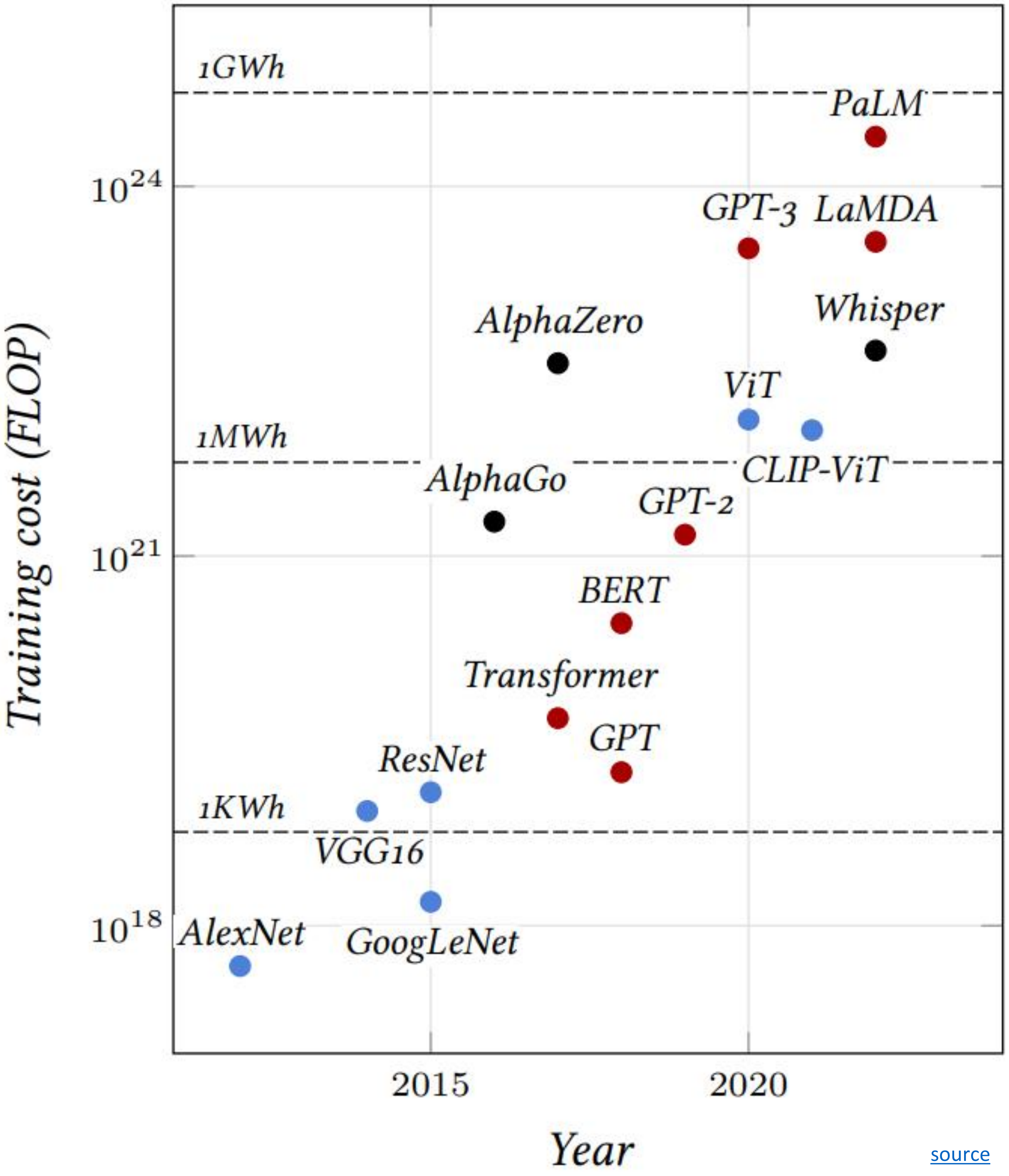


Text Generation

in-context learning as alternative to finetuning (new paradigm):
feed information into LLM via input prompt (decoder LLMs)
→ attention to context

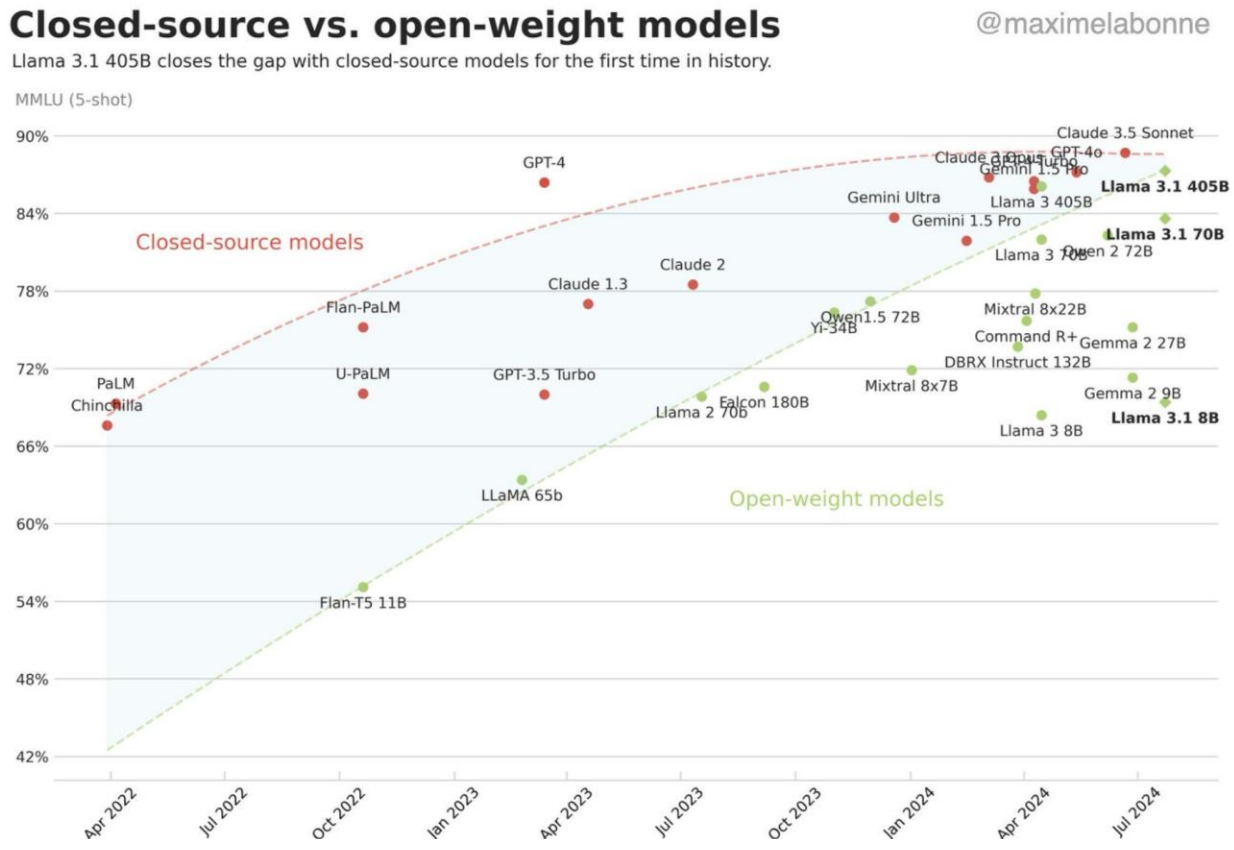
typical prompt:
instructions, context (potentially retrieved), query, output indicator

enables multi-task capabilities (all of ML before was only narrow tasks)
→ assistants



source

(open-source) SOTA (July 2024): Llama 3



a few surveys: 1, 2, 3

Some LLM Numbers

example Llama 3 405B:

- vocabulary size (tokens): 128K
- embedding/model dimensions: 16,384

- parameters: 405B
- training tokens: 15.6T

factor less than 40
→ a lot of memorizing

- context length/window (tokens): 128K

- training hardware: 16K GPUs (H100)

Decision Making

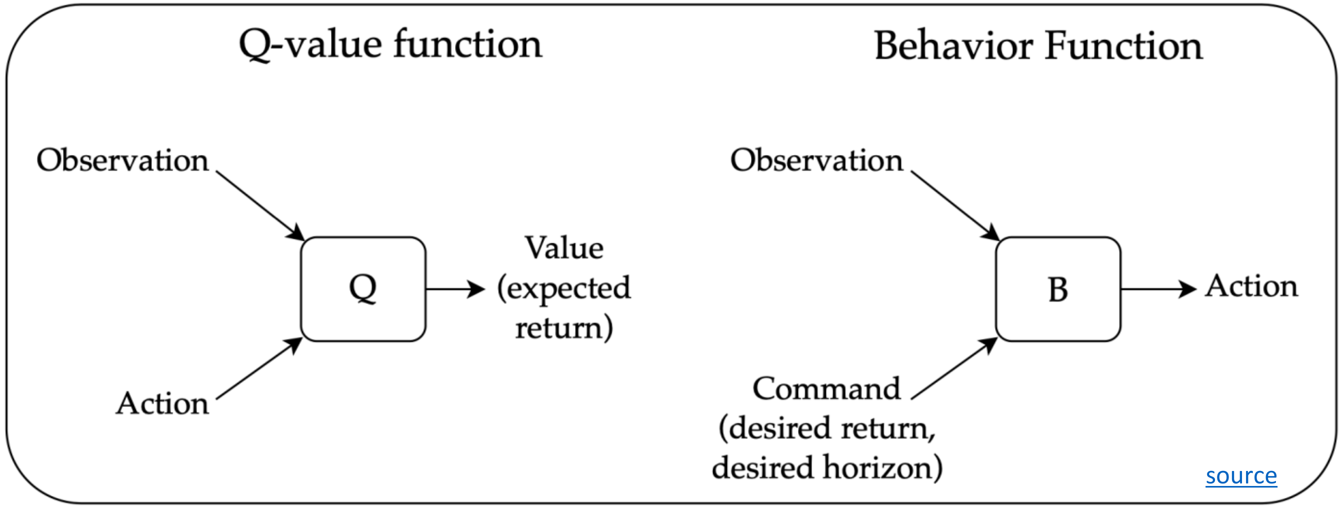
Sequential Decision Making

typically, domain of reinforcement learning (e.g., Q-learning or policy-gradient methods) or optimal control (e.g., model predictive control)

sequence modeling as alternative:

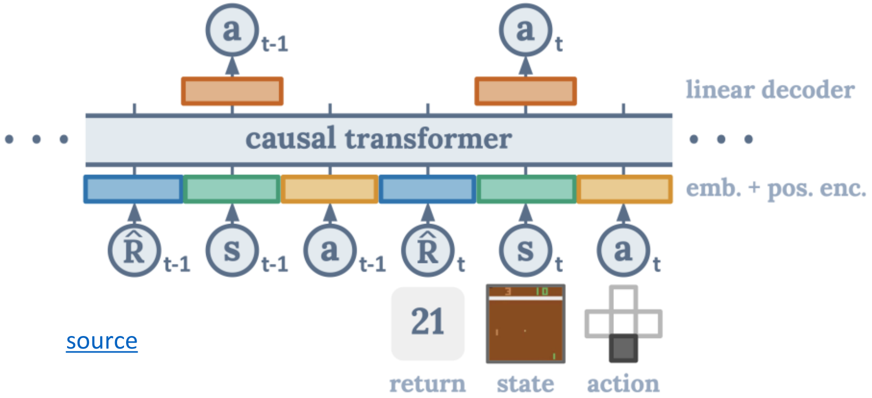
- generative: transformer to autoregressively model trajectories
- credit assignment directly via self-attention: state-return associations
- desired return tokens as prompt for action generation

overcoming the deadly triad of deep RL:



source

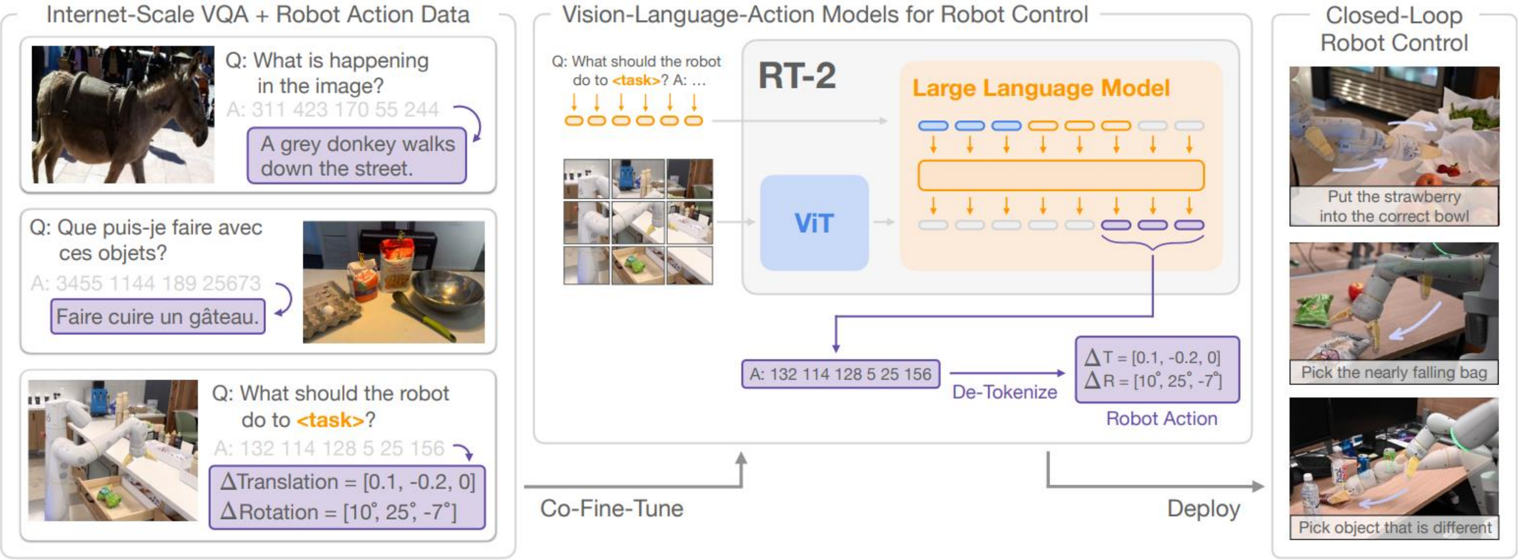
Decision Transformer:



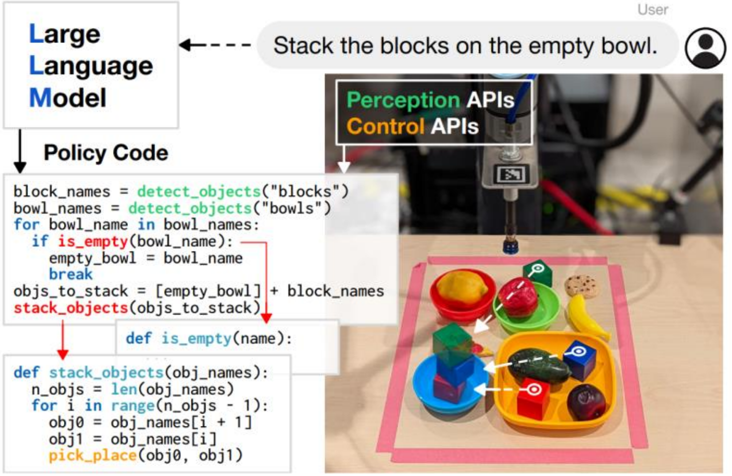
source

Robotic Control generated by LLMs/VLMs

RT-2:



Code as Policies:



SayCan (grounding with pre-trained skills):

(a) Large Language Models (LLMs)

(b) SayCan

Agents

LLM Agents

current AI good at learning statistical patterns and making predictions

but no real “understanding”, and limited reasoning and planning capabilities

desired agent capabilities:

- planning (LLM: decomposition of complex issue in multiple simple steps)
- tool use (LLM: use predictive models for numerical/optimization tasks)
- reflection
- collaboration with other agents

Goal: Autonomous End-to-End Workflow