

CHARLIE HEBDO

Journal irresponsable

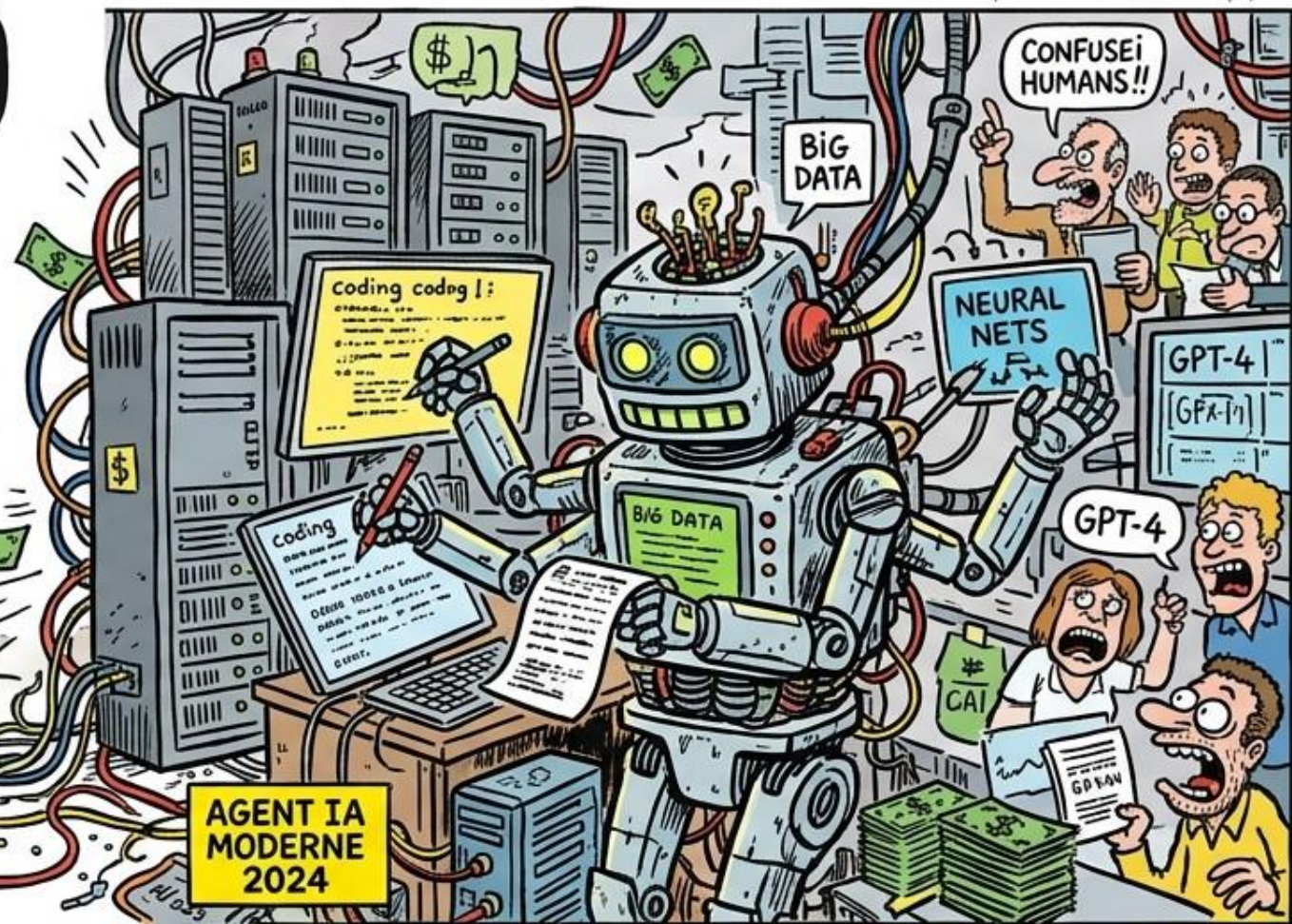
LES LLM

DE MARKOV AUX AGENTS IA

LE GRAND BORDEL DE L'INTELLIGENCE ARTIFICIELLE



CHAÎNES
DE MARKOV
1950



AGENT IA
MODERNE
2024

UNE LANGUE PEUT-ELLE PENSER?

#1 · Titre = page de couverture



La question philosophique qui dérange.

QUAND LE LANGAGE DEVIENT CALCUL

Je t'aime



$$\begin{pmatrix} 1 & 2 & 1 & 0 \\ 3 & 4 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{bmatrix} a_1 & a_2 & a_3 & -1 \\ a_{11} & a_{12} & a_{13} & -2 \\ 0 & -3 & 0 & 1 \end{bmatrix}$$

$$\rightarrow \left(\frac{a_{11} + 1}{2} \right) + \left(\frac{3 + 2}{\sqrt{14}} \right) = 1.89703301$$

La fin de la poésie, bonjour l'algorithme.

De Markov aux Agents IA : Les LLMs ou Comment un ~~modèle de~~ langage peut penser ?

Introduction

Imed MAGROUNE
septembre 2026

**1900 - 2012 :
PRÉHISTOIRE & IA SYMBOLIQUE**

MARKOV :
1^{ÈRE} MACHINE
À PRÉDIRE

SHANNON :
MESURE DE
L'INFORMATION

IA SYMBOLIQUE

PEUT-ELLE
PENSER ?

HIVERS DE L'IA



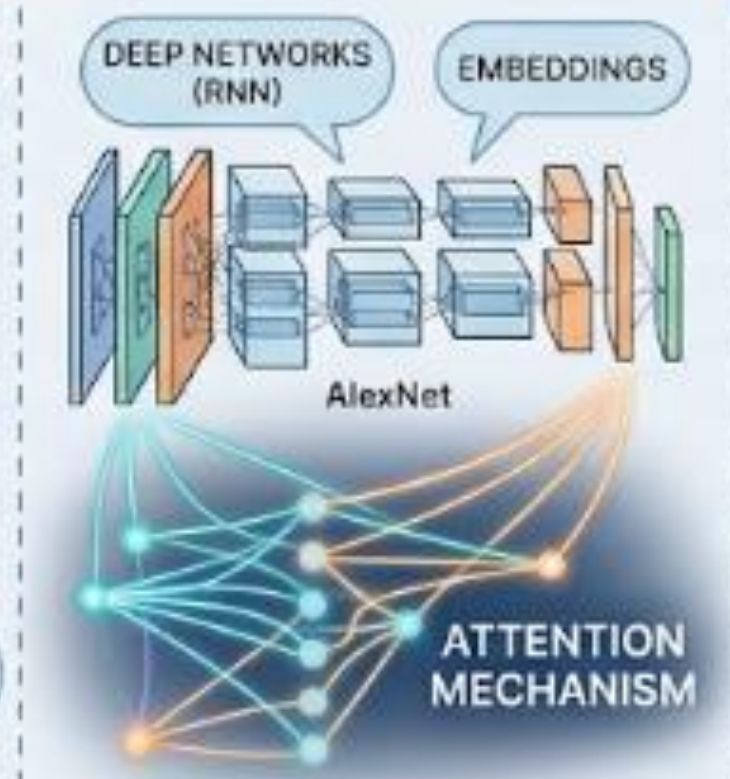
**2012 - 2017 :
L'AVÈNEMENT DU DEEP LEARNING**

DEEP NETWORKS
(RNN)

EMBEDDINGS

AlexNet

ATTENTION
MECHANISM



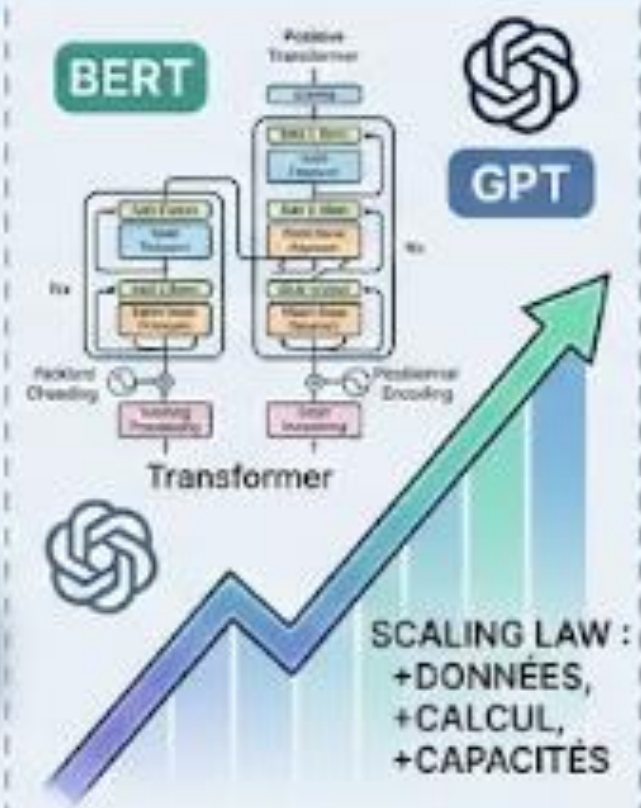
**2017 - 2022 :
L'ÉMERGENCE DU LLM**

BERT

GPT

Transformer

SCALING LAW :
+DONNÉES,
+CALCUL,
+CAPACITÉS



**2022 - 2026 :
LES AGENTS ET L'AUTONOMIE**

MULTIMODAL
(TEXTE, IMAGE,
AUDIO)

RAG

AI Agent

ELLE COMMENCE À AGIR

ELLE COMMENCE À AGIR



En 2026, l'IA n'attend plus une question : elle commence à agir.

Deux questions d'ouverture

« Je pense donc ... »

« Vous arriverez à lire ce texte, même si des lettres manquent et que certains mots sont mélangés. Vous n'avez pas besoin de tout voir : le contexte vous aide à compléter les blancs et à retrouver le sens. »



Le défi de Monty Hall



Trois portes : deux chèvres, une voiture.



Tu fais ton choix (33%)

*On ouvre une
autre porte
une chèvre !*



Veux-tu changer ton choix initial ?

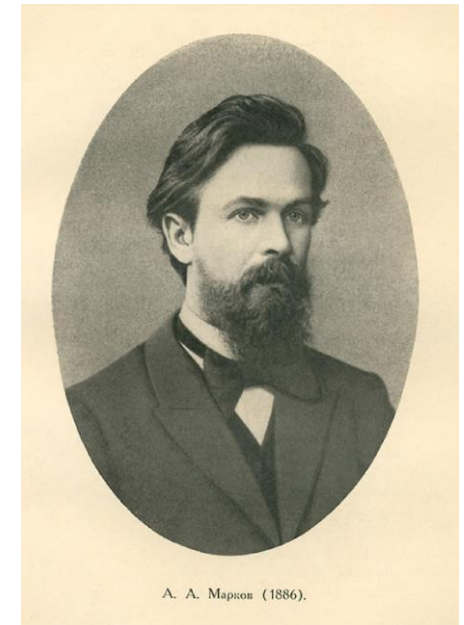
Prouver une loi, pas une croyance

Nekrasov, « le tsar des probabilités », contre Markov, le furieux



*Pavel Nekrasov
(1853–1924)*

*Andrei
Markov
(1856–
1922)*



Deux hommes, deux camps

Pavel Nekrasov — le théologien

- La loi des grands nombres n'a de sens que si les événements sont indépendants.
- L'indépendance des volontés serait la traduction du libre arbitre.
- Les mathématiques pourraient donc fonder la liberté — et Dieu.

Andrei Markov — le Furieux

- Les mathématiques n'ont rien à voir avec le libre arbitre ni avec la religion.
- Un théorème ne se croit pas : il se démontre.
- Il répondra par un contre-exemple, pas par la métaphysique. Du **texte**,

1913 — il attaque le langage

Il lui faut un terrain où la dépendance entre événements est évidente : un texte.

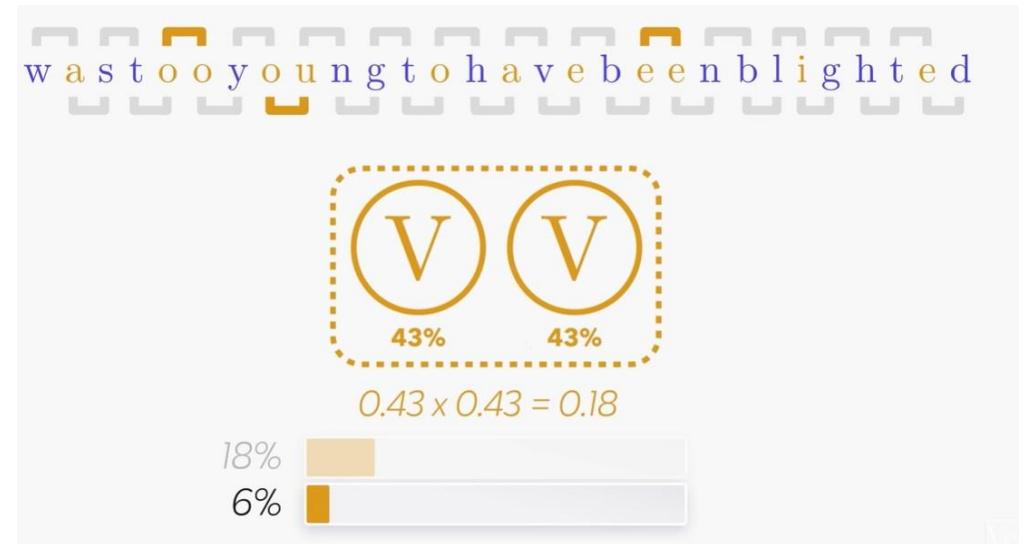
- Les 20 000 premières lettres d'Eugène Onéguine, le roman en vers d'Alexandre Pouchkine.
- Il jette les mots, la grammaire, le sens : chaque lettre devient voyelle (V) ou consonne (C).
- Reste une seule question mesurable : quelle lettre suit celle qu'on vient de lire ?



La preuve : les paires ne sont pas indépendantes

- Si les lettres étaient indépendantes :
 $43 \% \times 43 \% = 18 \%$ de paires VV.
- Dans Onéguine : 6 %. La dépendance est mesurée, pas supposée.
- Quatre paires, quatre probabilités

W 5,5 % · VC 37,7 % · CV 37,7 % · CC 19,1 %

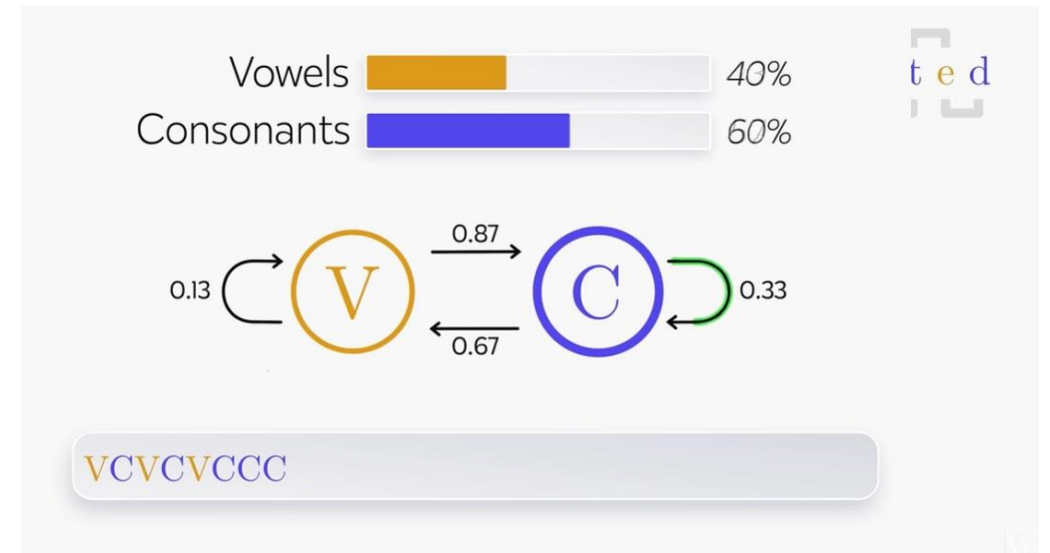


18 % de paires VV si les lettres étaient indépendantes ($0,43 \times 0,43$) contre 6 % observés : la dépendance est mesurée

La machine à prédire

- Après une voyelle : 13 % une voyelle, 87 % une consonne.
- Après une consonne : 67 % une voyelle, 33 % une consonne.
- Quatre nombres, et la machine tire la lettre suivante selon la probabilité de l'état courant, puis recommence.
- Elle ne connaît ni Pouchkine, ni les mots, ni le sens : le texte produit obéit aux fréquences mesurées à la main.

#4 · Markov (1906 → 1913) : le langage devient probabilités



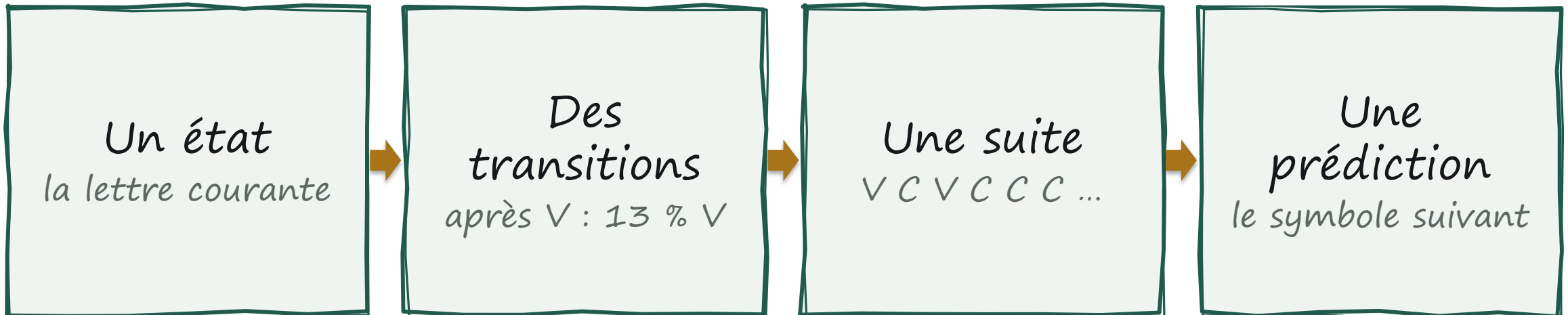
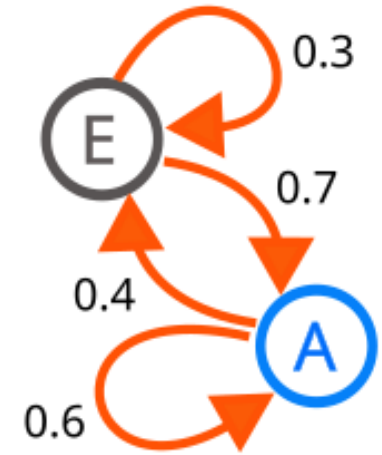
Après une voyelle : 13 % V / 87 % C · après une consonne : 67 % V / 33 % C — quatre nombres suffisent à générer une séquence

« On peut avoir du hasard avec de la dépendance — et une moyenne qui converge quand même. »

— Ce que Markov démontre en 1906, puis mesure sur Pouchkine en 1913

La chaîne de Markov, c'est quoi ?

Des états et des transitions qui dépendent uniquement de l'état courant.



De Markov à "l'algorithme" qui prédit presque tout

- Décrire le monde par un état, puis modéliser les transitions possibles vers l'état suivant.
- Cette idée structure les processus de décision, la recherche sur le Web, la planification et l'apprentissage par renforcement.
- Aujourd'hui, les modèles de langage prédisent eux aussi la suite — mais à partir d'un contexte appris, beaucoup plus riche.

État présent → futurs possibles → probabilités

Le graphe devient le Web



Mêmes états, mêmes transitions — un autre monde.



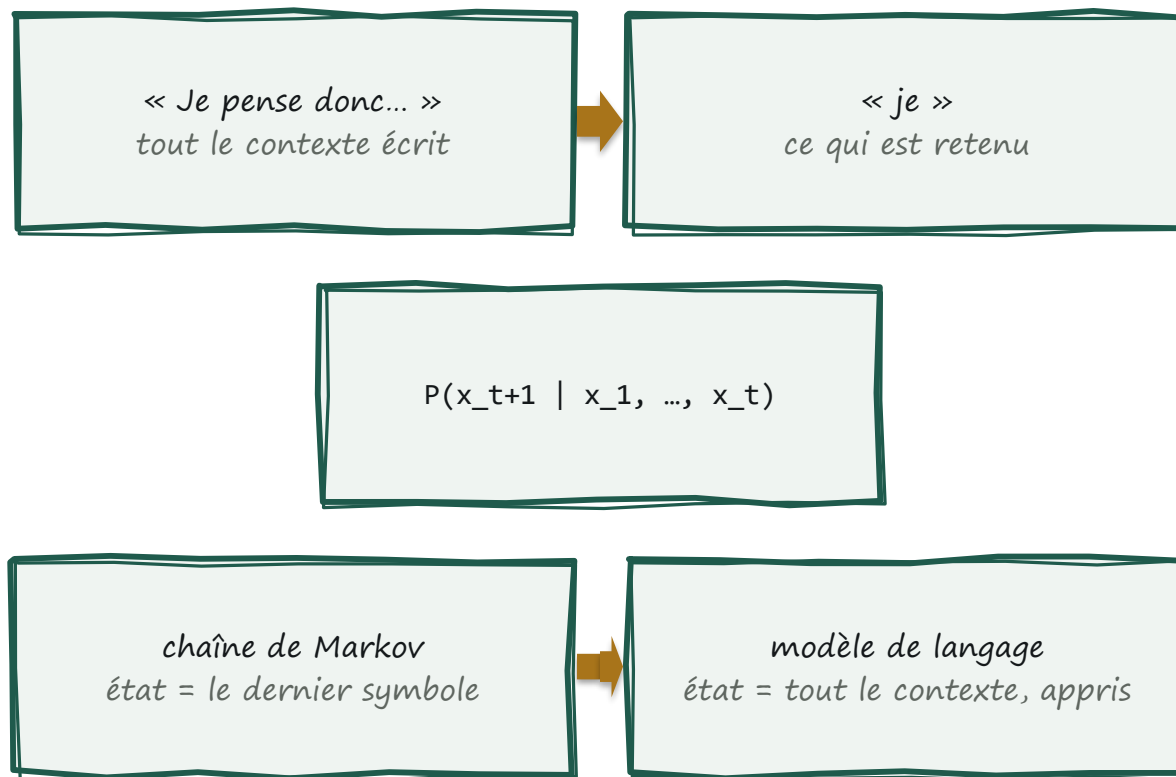
Le graphe devient un arbre de décisions

Les transitions deviennent des actions ; aux feuilles, des scores.



MCTS n'est pas une chaîne de Markov : c'est une méthode de recherche dans un arbre, souvent appliquée à des problèmes modélisés comme des MDP.

Les états deviennent des tokens



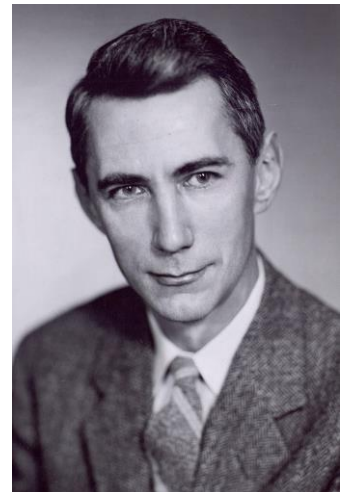
Flèche en pointillés : filiation conceptuelle, pas une dérivation mathématique.

Et ensuite ? De la prédiction à la question de la pensée

*En 1950, Turing pose la question ; en 1951, Shannon
en fait une mesure*



*Alan Turing
(1912-1954)*



*Claude
Shannon
(1916-
2001)*

Turing, 1950 : can machines think ?

Avant de demander si une machine peut penser, Turing théorise en 1936 la machine qui calcule.

1936 : la machine de
Turing
une bande de cases, une tête
qui lit et écrit

CAN MACHINES
THINK? (1950)

la question, posée sans définir
« penser »

The Imitation Game

un juge, deux interlocuteurs,
des questions écrites

Parler = l'interface

on ne teste pas l'esprit : on
teste la conversation

1951 : mesurer la
structure du langage

après Turing, Shannon passe
aux statistiques

Le papier : Computing Machinery and Intelligence (1950)

- A. M. Turing, *Mind*, vol. 59 — octobre 1950
- La question « une machine peut-elle penser ? » est remplacée par un jeu
- Ce qu'on mesure : une conversation indiscernable, pas une conscience

#6 · Turing, 1950 : Can Machines Think ?

A. M. Turing (1950) Computing Machinery and Intelligence. *Mind* 49: 433-460.

COMPUTING MACHINERY AND INTELLIGENCE

By A. M. Turing

1. The Imitation Game

I propose to consider the question, "Can machines think?" This should begin with definitions of the meaning of the terms "machine" and "think." The definitions might be framed so as to reflect so far as possible the normal use of the words, but this attitude is dangerous. If the meaning of the words "machine" and "think" are to be found by examining how they are commonly used it is difficult to escape the conclusion that the meaning and the answer to the question, "Can machines think?" is to be sought in a statistical survey such as a Gallup poll. But this is absurd. Instead of attempting such a definition I shall replace the question by another, which is closely related to it and is expressed in relatively unambiguous words.

The new form of the problem can be described in terms of a game which we call the 'imitation game.' It is played with three people, a man (A), a woman (B), and an interrogator (C) who may be of either sex. The interrogator stays in a room apart from the other two. The object of the game for the interrogator is to determine which of the other two is the man and which is the woman. He knows them by labels X and Y, and at the end of the game he says either "X is A and Y is B" or "X is B and Y is A." The interrogator is allowed to put questions to A and B thus:

C: Will X please tell me the length of his or her hair?

Now suppose X is actually A, then A must answer. It is A's object in the game to try and cause C to make the wrong identification. His answer might therefore be:

"My hair is shingled, and the longest strands are about nine inches long."

In order that tones of voice may not help the interrogator the answers should be written, or better still, typewritten. The ideal arrangement is to have a teleprinter communicating between the two rooms. Alternatively the question and answers can be repeated by an intermediary. The object of the game for the third player (B) is to help the interrogator. The best strategy for her is probably to give truthful answers. She can add such things as "I am the woman, don't listen to him!" to her answers, but it will avail nothing as the man can make similar remarks.

We now ask the question, "What will happen when a machine takes the part of A in this game?" Will the interrogator decide wrongly as often when the game is played like this as he does when the game is played between a man and a woman? These questions replace our original, "Can machines think?"

*Computing Machinery and
Intelligence — A. Turing,
Mind, 1950*

Shannon, 1951 : quelle sera la prochaine lettre ?

1948 : Shannon fonde la théorie de l'information.



1948 : la théorie de
l'information

*l'information, c'est
l'imprévisible*

1950 : Theseus

*une souris mécanique qui
apprend un labyrinthe*

1951 : prédire la
lettre suivante

*le langage comme suite de
probabilités*

1 → 100 lettres de
contexte

*plus de contexte, moins
d'incertitude*

La redondance du
langage

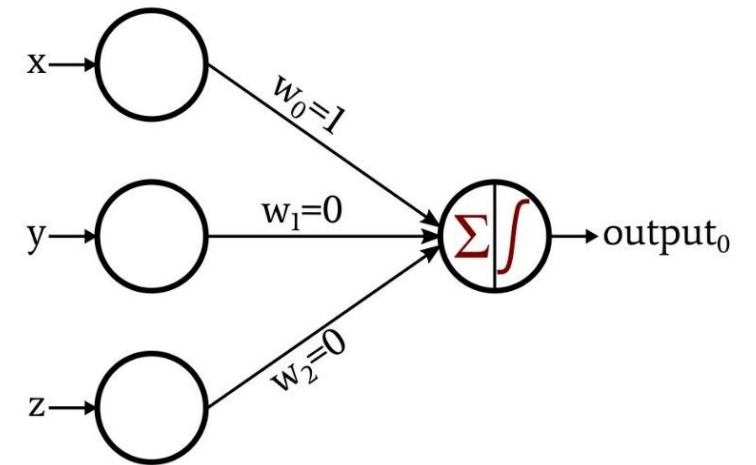
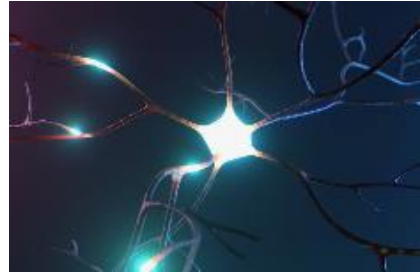
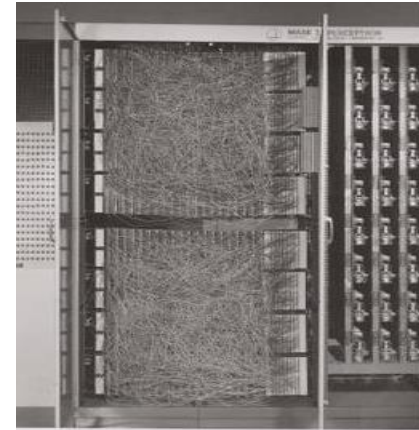
*le langage en dit plus qu'il n'en
faut*

L'IA symbolique VS Réseaux de neurones

- Systèmes experts et règles écrites à la main
- Arbres de décision
- Features conçues à la main
- NLP symbolique et statistiques classiques

1958 — le perceptron, la première machine qui apprend

- Frank Rosenblatt : des poids ajustés à partir d'exemples
- Une seule couche de neurones, une règle de correction
- Une promesse immense... et beaucoup de bruit médiatique



1997 — Deep Blue bat Kasparov



- Une victoire historique... de l'IA SYMBOLIQUE
- Environ 200 millions de positions évaluées par seconde
- Des critères d'évaluation conçus par des experts
- Une IA spécialisée dans les échecs, sans réseau neurones



Deux voies, deux philosophies

Symbolique — écrire les règles

- Des règles et des connaissances écrites à la main
- Explorer un espace de recherche
- Gagner aux échecs (Deep Blue, 1997)

Neuronal — apprendre les poids

- Ajuster des poids sur des exemples
- Des représentations apprises, pas données
- Longtemps... loin du niveau requis

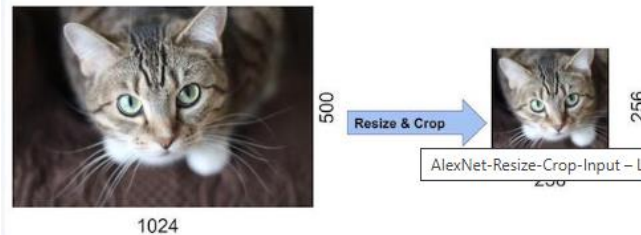
Des décennies de résultats médiocres

- La rétropropagation existe depuis 1986
- Mais les données et la puissance de calcul manquent
- En vision comme en langage, les méthodes à la main dominant
- Il faudra attendre 2012 — et un autre ingrédient

AlexNet, 2012 : la représentation devient apprise

ImageNet : 1,2 million d'images, un réseau profond, des GPU, 60 millions de paramètres.

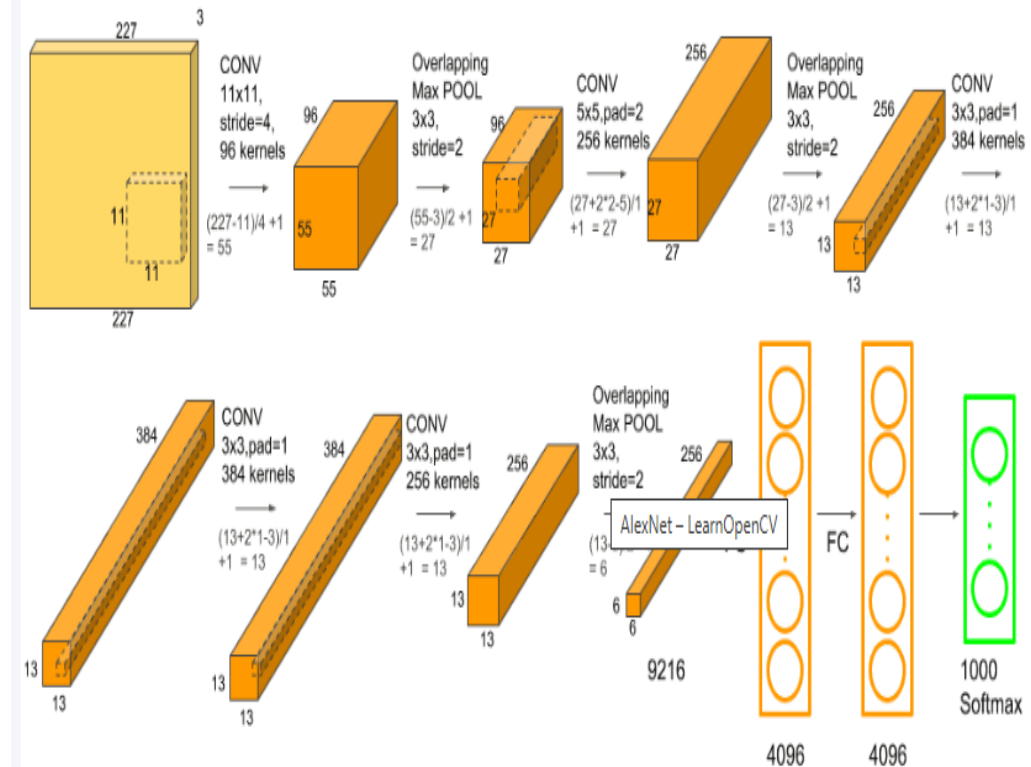
Image brute
des pixels, rien d'autre



Couches
convolutions, puis couches
denses

Représentations
contours, textures, objets
— apprises, pas écrites

Classification
et la performance en vision
bascule

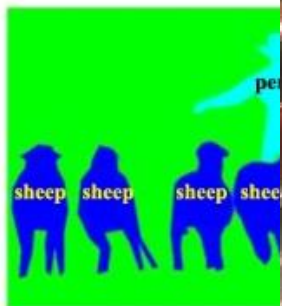


Du visible au texte : les deux mondes se rejoignent

Combien de vraies voitures voyez-vous ?



(a) Object Cl



(c) Semantic S



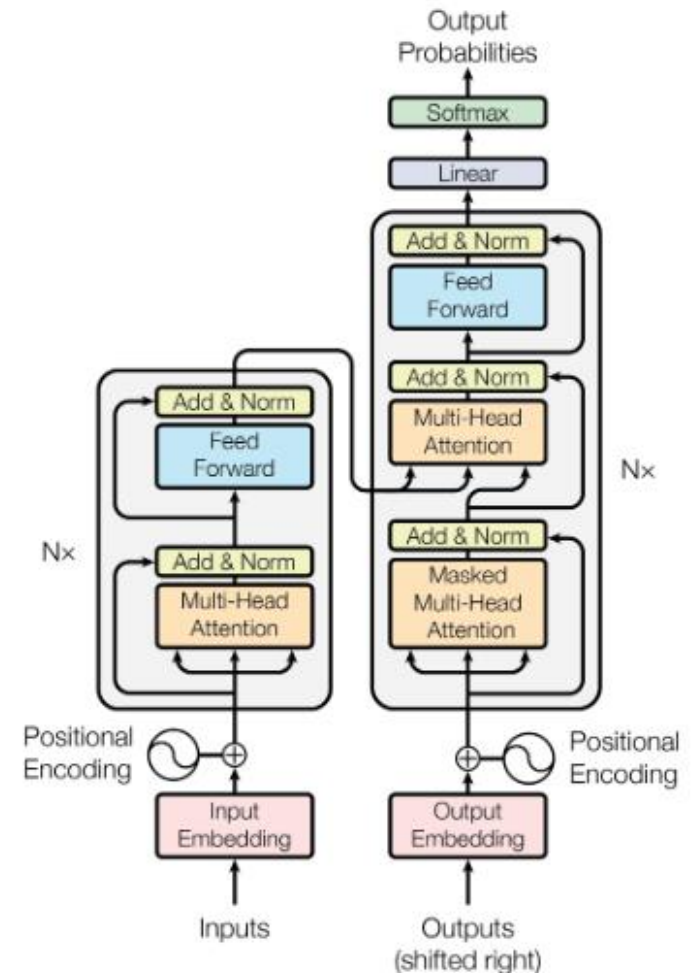
Image captioning : la vision et le langage se parlent

- 2014-2015 : un réseau convolutif regarde, un modèle de langue décrit
- Une image entre, une séquence de mots sort
- C'est le geste même d'un LLM : encoder une entrée, générer une sortie



Alors la question change

- Comment aligner une séquence sur une autre ?
- Comment décider QUE regarder en écrivant chaque mot ?
- Réponse, en 2017 : l'attention



Le problème des données

- image → label
- texte → label
- Internet, livres, code, conversations, vidéos... : impossible à annoter à la main

L'auto-supervision : le texte est son propre professeur

Aucune étiquette à écrire : la phrase qui suit est la bonne réponse.

« La Terre tourne autour
du... »
on montre le début, on cache
la suite

Masquer la suite

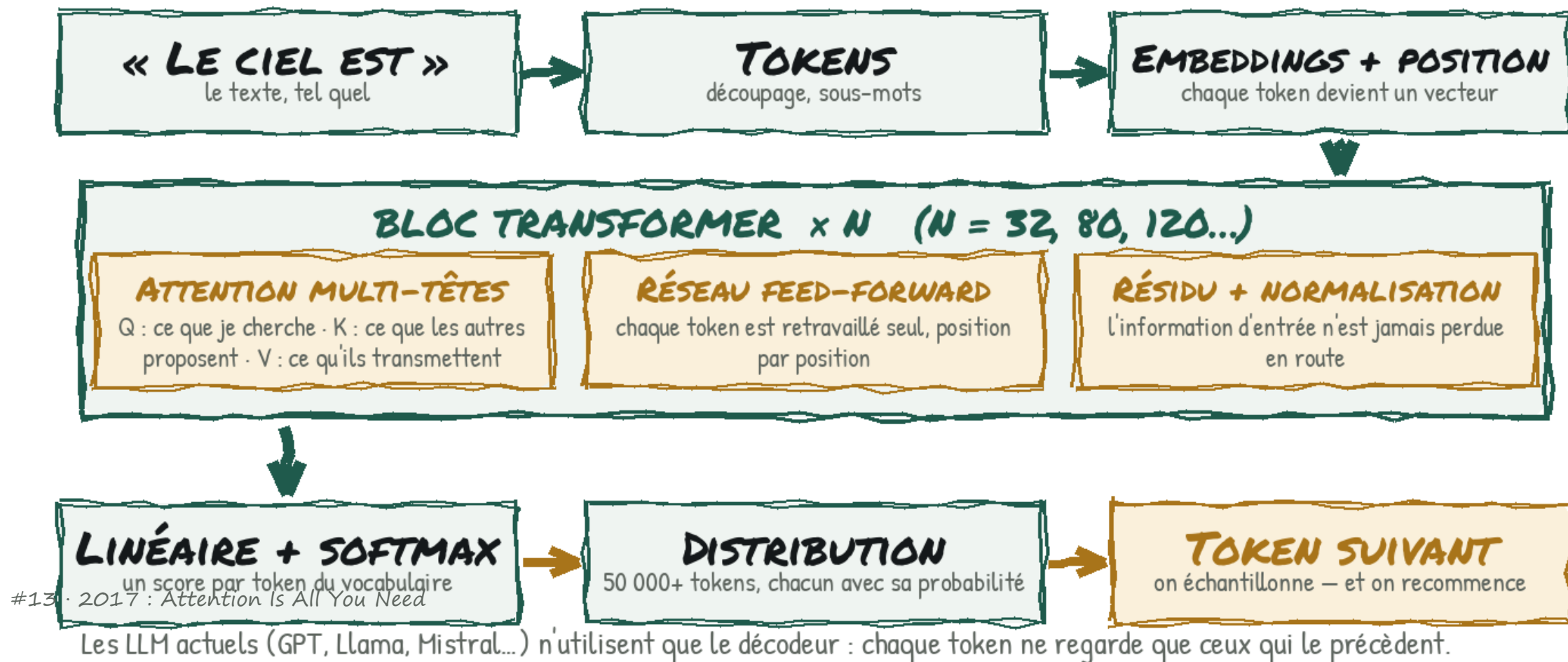
Prédire
« ... du Soleil »

Erreur → gradient

× des milliards
chaque phrase d'Internet
devient un exercice corrigé

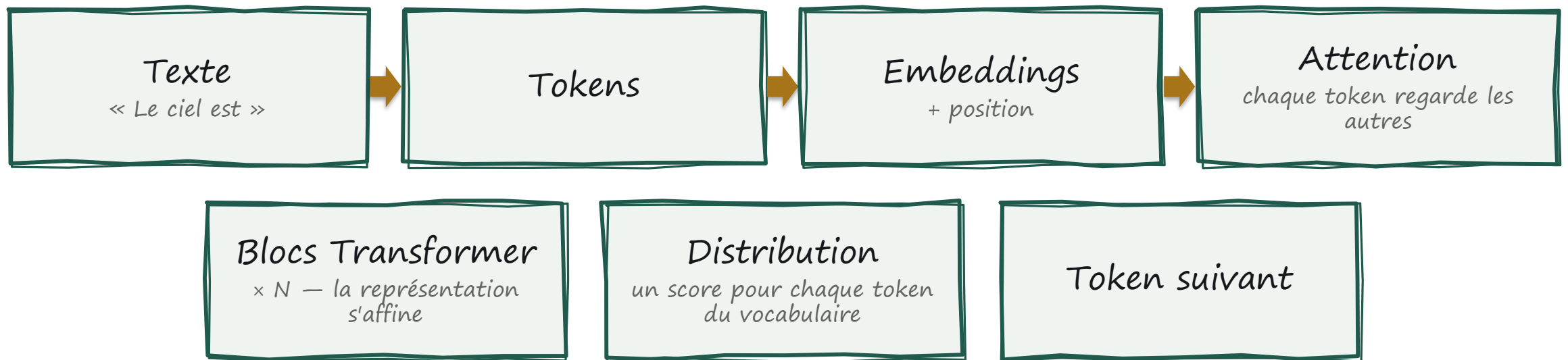
LE TRANSFORMER

« Attention Is All You Need » — Vaswani et al., 2017 (arXiv:1706.03762)



Le chemin d'un texte dans le Transformer

Le Transformer abandonne la récurrence : tout se calcule en parallèle, l'attention fait circuler l'information.



L'attention : le moment où le contexte apparaît

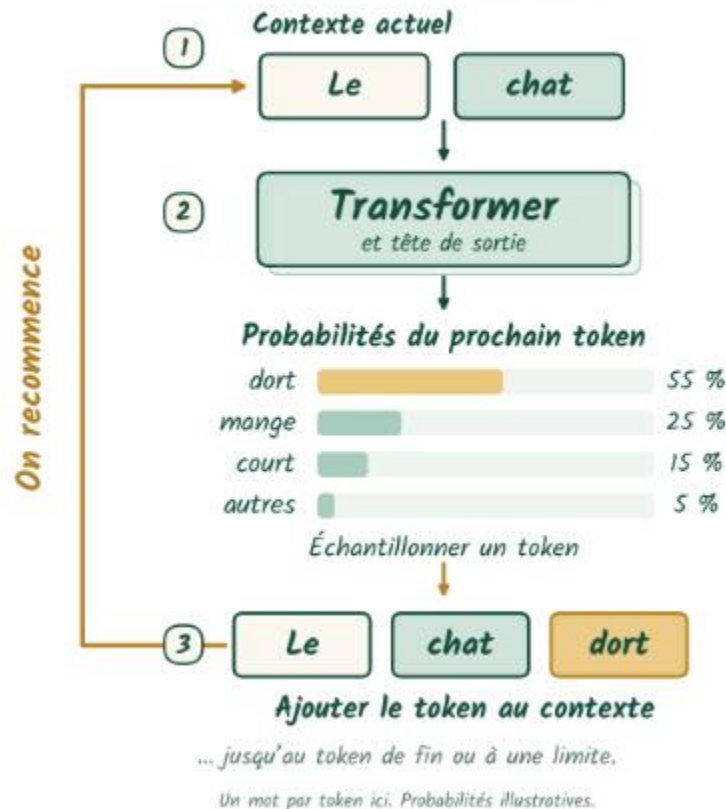
« The bank was crowded because it was payday. » — le token « bank » regarde le reste de la phrase.



Qu'est-ce qu'un LLM apprend réellement ?

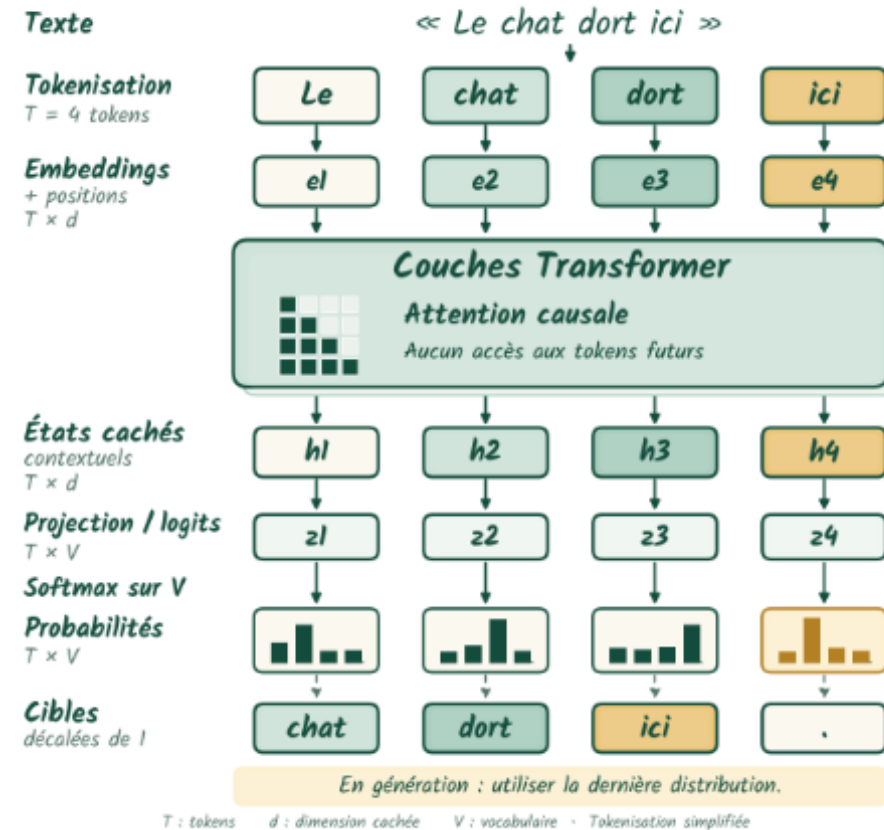
Générer un token à la fois

La sortie devient la nouvelle entrée



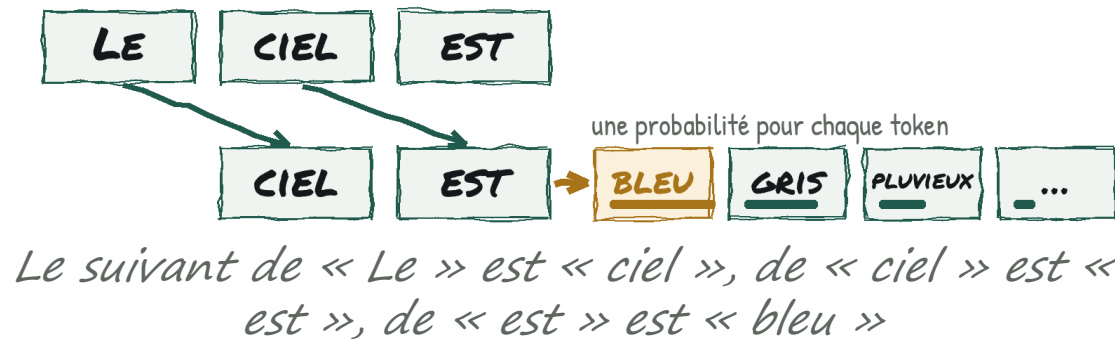
Apprendre sur toute la séquence

Chaque position prédit le token suivant



Le pré-entraînement : prédire, encore et encore

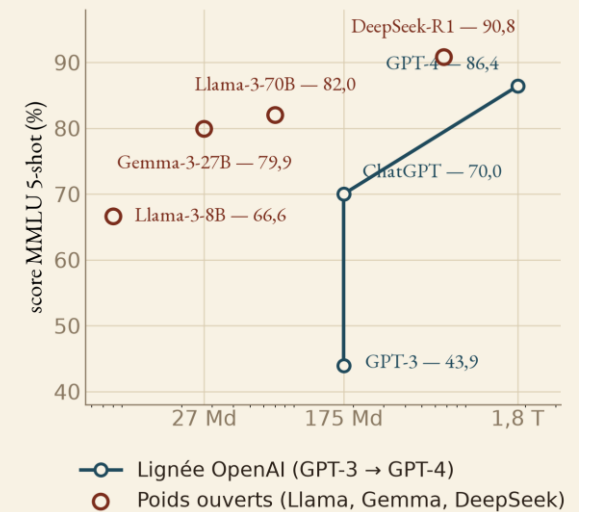
- On lui donne d'immenses quantités de texte, et une seule consigne : le token suivant
- Le texte est sa propre leçon : aucune réponse attendue, aucune étiquette
- Il en sort la grammaire, les faits, les styles — et un peu de raisonnement



Le modèle de fondation

- Un modèle pré-entraîné, généraliste, réutilisable pour mille usages
- Très coûteux à produire une fois, très bon marché à adapter ensuite
- Distribué par API, ou en poids ouverts (Llama, Mistral...)
- Ce n'est pas encore un interlocuteur : c'est une matière première

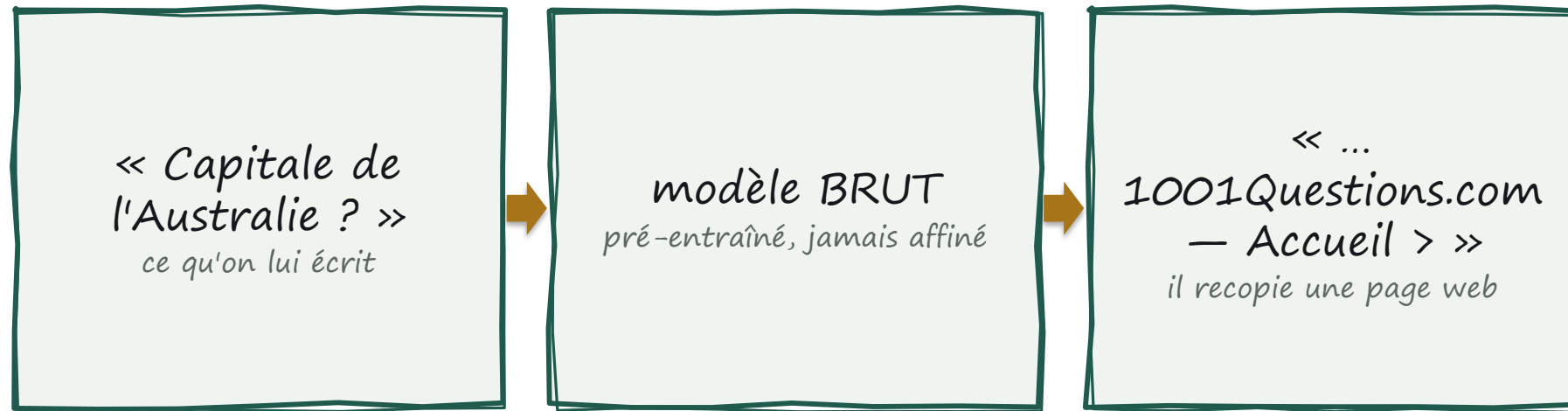
Performances et taille des modèles



MMLU 5-shot, valeurs publiées (2020-2025) · échelle des tailles logarithmique

*MMLU 5-shot — valeurs
publiées par les laboratoires
(2020-2025)*

Le modèle brut ne répond pas : il continue



Il a appris la forme des questions, pas qu'on attend une réponse.

Le fine-tuning : apprendre la forme de l'échange

- On continue l'entraînement sur des exemples de dialogues réussis
- Chaque exemple : une demande, et la réponse attendue
- Le modèle ajuste ses poids pour répondre, plus pour continuer
- Le modèle brut devient un modèle « instruct » ou « chat » ...

Pré-entraînement, SFT, RL, humain feedback, PPO

Chaque étage répond à un manque du précédent.



Sources : SFT/instruction tuning — FLAN (2021) et InstructGPT (2022) ·
retour humain (RLHF) — Christiano et al. (2017)

Le gabarit du prompt (SFT)

rôle	ce qu'il contient	qui l'écrit
system	le cadre : ton, règles, outils autorisés	l'application
user	la demande de l'utilisateur	l'utilisateur
assistant	la réponse attendue, et les suivantes	le modèle

Le gabarit est reconstruit à chaque tour : le modèle ne se souvient pas, on lui réécrit tout le contexte.

Le tool calling : répondre par une action

- Le modèle apprend à produire un appel structuré : un outil, des arguments
- Le programme exécute l'outil et renvoie le résultat au modèle
- Exemple : « quelle heure à Sydney ? » → appel de l'horloge, puis réponse
- C'est ce qui transforme un modèle en agent

Apprendre sans toucher aux poids

In-context learning — Brown et al., 2020 (GPT-3)

- Quelques exemples dans le prompt changent le comportement
- Zéro, un ou quelques exemples : le modèle « apprend » à la volée
- Aucun poids modifié : le contexte fait tout le travail

Chain-of-thought — Wei et al., 2022

- « Raisonne étape par étape » avant de conclure
- Les étapes intermédiaires servent de brouillon
- Réfléchir coûte des tokens : on génère plus pour décider mieux

In-context learning : « Language Models are Few-Shot Learners », arXiv:2005.14165 (NeurIPS 2020)
• *Chain-of-thought* : « Chain-of-Thought Prompting Elicits Reasoning in Large Language Models », arXiv:2201.11903 (NeurIPS 2022)

Les artefacts de l'apprentissage

Le modèle apprend aussi...

- les biais du corpus
- ses erreurs et contradictions
- ses stéréotypes
- ses conventions et ses styles
- certaines données mémorisées
- les défauts de qualité du corpus

Pourquoi le LLM hallucine

Le modèle ne cherche pas la vérité : il continue le texte le plus plausible.

« Qui a inventé X
en 1847 ? »

*une question dont la réponse
n'est pas dans ses données*

**Continuation
plausible**

*la suite la plus probable,
question incluse*

Affirmation assurée

le ton est celui d'un fait vérifié

Réalité : faux

**CONFIDENCE ≠
TRUTH**

la fluidité n'est pas une preuve

Les questions qui surgissent

- Données : qui les possède ? a-t-on le droit de les utiliser ?
- Propriété intellectuelle : mémorisation, reproduction, attribution
- Entreprise : données confidentielles, secrets, fuites de contexte
- Sécurité : prompt injection, exfiltration, actions non désirées
- Biais : le modèle apprend ceux présents dans les données

Du modèle au système

Un modèle seul est une brique : ce qui en fait un produit, c'est ce qu'on met autour.

LLM seul

*une conversation, sans
mémoire*

+ contexte :
RAG

*on lui fournit les
documents pertinents*

+ mémoire

*l'assistant se souvient
des tours précédents*

+ outils

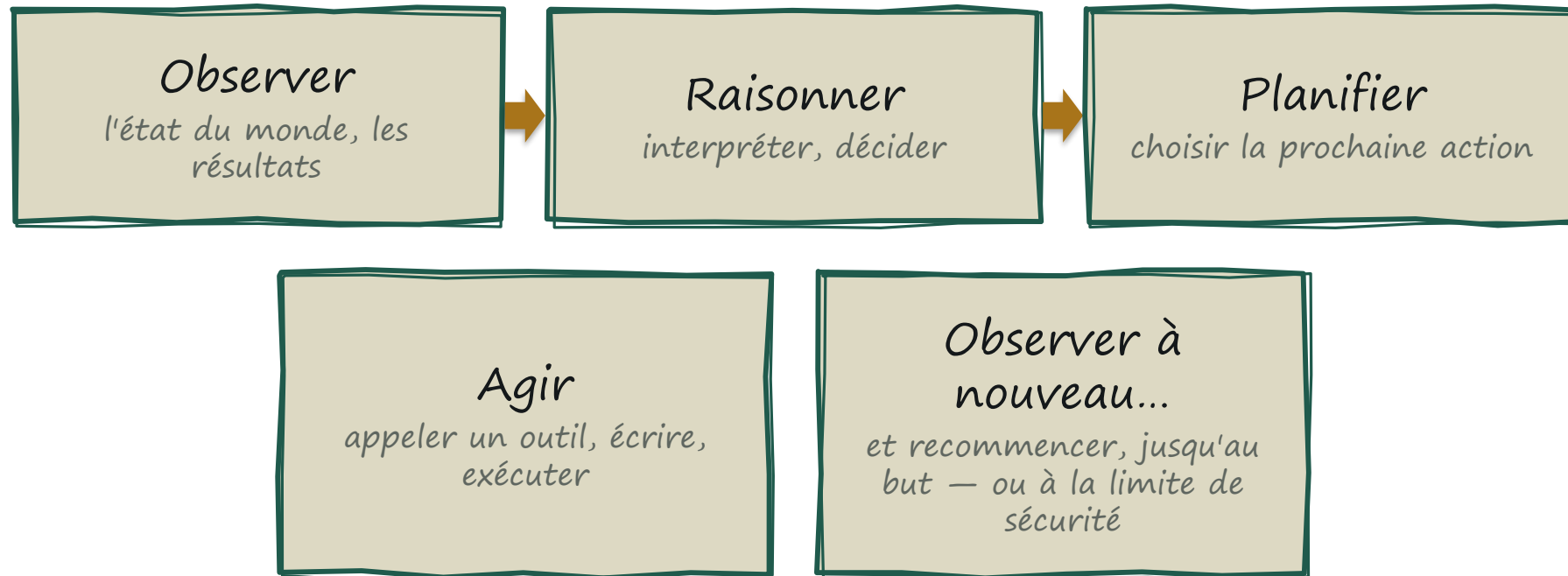
*il peut agir : API,
fichiers, calcul*

+ boucle :
l'AGENT

*il observe le résultat et
décide de la suite*

La boucle agentique

Un agent n'est pas un modèle plus gros : c'est une boucle autour du même modèle.



Un exemple spectaculaire : Moltbot → OpenClaw

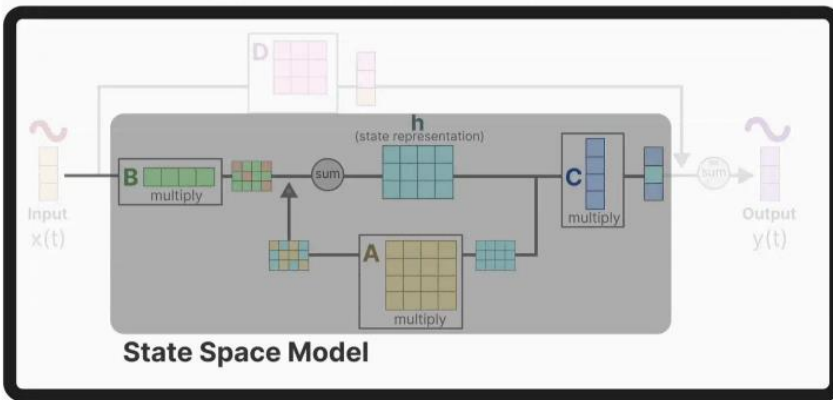
OPENCLAW

assistant personnel open-source (ex-Moltbot) — un agent qui agit

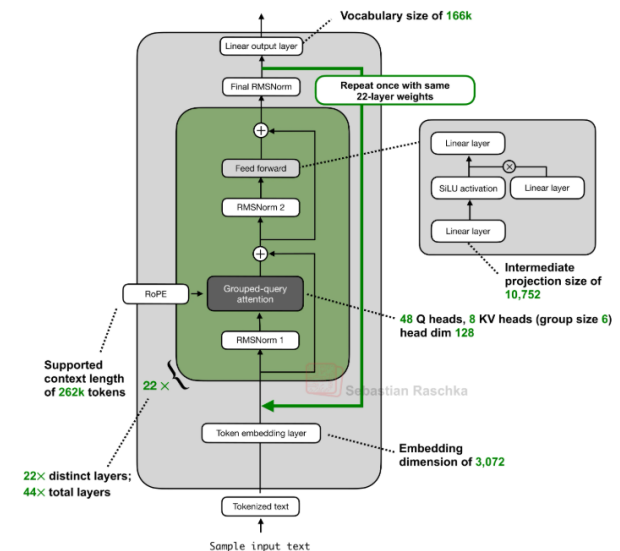


Autres architectures

🦜🔗 Llama + SSMs



Nanbeige 4.2 (3B)



quelques exemples , il y en a d'autres

Le paradoxe multimodal

Le même mécanisme qui prédit des mots traite aussi des images et du son.



Comment sait-on qu'un modèle est bon ?

Un benchmark, c'est quoi ?

- Un jeu de questions, leurs réponses, et un protocole de notation
- Un but simple : comparer des modèles sur exactement la même épreuve
- Un piège immédiat : mesurer la mémoire du test plutôt que la capacité

Les épreuves les plus connues

<i>épreuve</i>	<i>ce qu'elle teste</i>	<i>format</i>	<i>statut 2026</i>
MMLU / MMLU-Pro	culture générale	QCM, 57 matières	saturé (~94 %)
GPQA Diamant	science niveau thèse	448 questions	proche du plafond (93-96 %)
SWE-bench Vérifié	vrai travail de code	500 tickets GitHub	relayé par SWE-bench Pro
AIME	maths de compétition	15 réponses entières	saturé (abandonné)
ARC-AGI-2	raisonnement abstrait	grilles à transformer	en cours (77-92 %)
Humanity's Last Exam	savoir expert, 100+ domaines	2 500 questions	le discriminant (13-65 %)
Arène (ELO)	préférence humaine	duels en aveugle	le classement vivant

Un score n'a de sens qu'à réglage égal : outils, budget de raisonnement et programme d'évaluation changent le résultat.

Comment un score se calcule

- On extrait la réponse, puis on la compare : lettre, entier, correspondance exacte
- Pour le code : la correction proposée passe — ou non — les tests du projet
- Pour le texte libre : un juge note, un modèle ou un humain
- Température, nombre d'essais, budget de raisonnement : tout cela fait partie du score

L'ELO des arènes : classer sans corriger

- On ne note pas une réponse : on en préfère une parmi deux, en aveugle
- Chaque duel ajuste la note des deux modèles — c'est le classement ELO
- Chatbot Arena : des centaines de milliers de votes (texte, code, vision)
- ± 5 à 8 points d'incertitude — et « préféré » ne veut pas dire « juste »

Trois pièges qui changent tout

- **Contamination** : les questions publiques finissent dans les données d'entraînement
- **Saturation** : MMLU, AIME et GPQA sont au plafond — elles ne départagent plus
- **Harnais** : le même modèle, deux scores selon les outils et le raisonnement autorisé
- **Règle de lecture** : moins de 3 points d'écart, ce n'est pas une différence

2026 : de la conversation au travail

Les modèles classés par capacités — 1/2

- Raisonnement : modèles frontier généralistes
- Coding : agents avec terminal, IDE, navigateur, dépôts
- Computer use : le modèle voit l'interface et agit
- Recherche : navigation + raisonnement + synthèse

Les modèles classés par capacités — 2/2

- Agents : mémoire + outils + planification + boucles
- Multimodal : texte + vision + audio + vidéo
- Edge : smartphones, PC, Raspberry Pi, Jetson, robots

« Quel modèle est suffisamment bon
pour CETTE tâche, à ce coût, cette
latence, ce niveau de confidentialité ?

>>

— la bonne question — pas « quel est le meilleur
modèle ? »

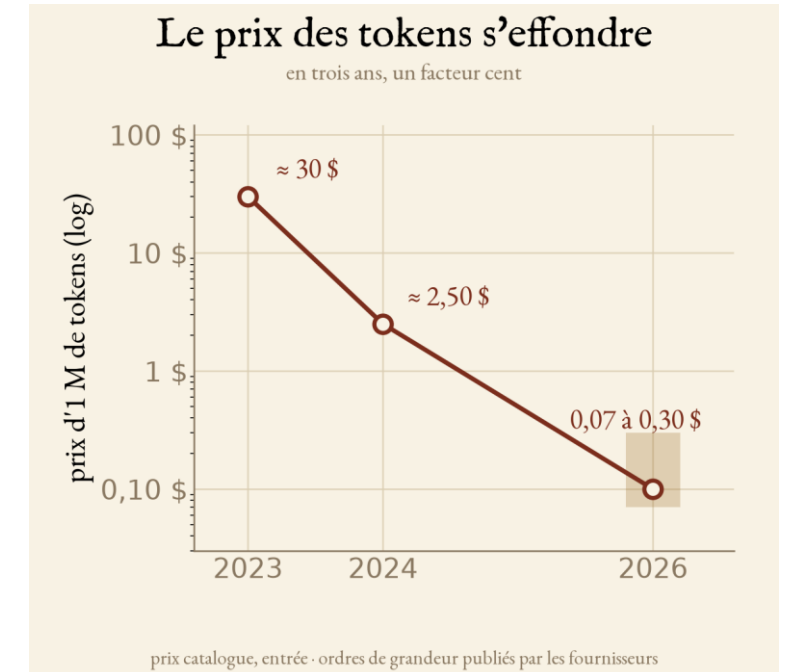
Du chatbot à l'agent

- 2022 : un chatbot répond à une question
- 2026 : un agent exécute la tâche de bout en bout — terminal, navigateur, dépôts
- Mémoire + outils + planification + boucle de vérification
- Ce n'est plus seulement le modèle qui change : c'est le système autour (Suite du Web 3,0 et de Software 3,0)

Les prix de l'API s'effondrent

- 2023 — modèle frontier : ~ 30 \$ le million de tokens en entrée
- 2024 — même classe : $\sim 2,50$ \$ (-90 %)
- 2026 — même qualité : 0,07 à 0,30 \$ (-99 %)
- Moteurs : concurrence, poids ouverts, optimisation d'inférence

#25 · Le SOTA 2026 : une carte des capacités



Prix catalogue d'1 M de tokens en entrée — ordres de grandeur (2023-2026)

L'open source rattrape

- Des poids ouverts au niveau de la frontière
- Auto-héberger devient une option réelle
- Souveraineté, coût, confidentialité : trois arguments

LES FAMILLES DE MODÈLES (2026)

celles que nous croiserons dans le cours

 Qwen Alibaba — la famille ouverte la plus large	 DeepSeek raisonnement ouvert, poids publiés
 Gemma Google DeepMind — du téléphone au poste	 Granite IBM — dense 3, 8 et 30 Md, contexte 512K

Les familles de modèles (2026) — fermés et/ou poids ouverts

Les modèles frontière en septembre 2026

modèle	Humanity's Last Exam	GPQA Diamant	code	prix / M tokens
Claude Fable 5.1 — Anthropic	65,0 % (outils)	92,6 %	95 % SWE Vérifié	10 \$ / 50 \$
GPT-6 Astra — OpenAI	57,2 % (outils)	96,0 %	57,7 % Terminal-Bench	10 \$ / 50 \$
Gemini 3.8 Flash — Google	54,9 %	—	—	3,75 \$
DeepSeek V4-Pro — ouvert	60,0 % (outils)	90,1 %	80,6 % SWE Vérifié	0,43 \$ / 0,87 \$
GLM-5.2 — Zhipu, ouvert	54,7 % (outils)	91,2 %	62,1 % SWE Pro	1,40 \$ / 4,40 \$
Grok 4.6 — xAI	42,9 %	94,9 %	—	—

Valeurs publiées par les laboratoires, juin-septembre 2026 ; le réglage compte — « avec outils » n'est pas comparable à « sans outils ».

Ce que le tableau dit vraiment

- Le meilleur n'est pas le même selon l'épreuve : savoir, science, code
- Sur le savoir expert, tous restent entre 13 et 65 % : l'examen n'est pas passé
- Les poids ouverts sont à quelques points des fermés, pour dix à cent fois moins cher

Les modèles de 2026 : noms, tailles, perfs

Du data center à la carte à 80 €

Cinq familles, deux mondes

- Frontier propriétaires : ChatGPT, Gemini, Claude
- Open-weight : Qwen, Llama, DeepSeek, Gemma
- La frontière entre les deux s'amincit chaque mois

LES FAMILLES DE MODÈLES (2026)

celles que nous croiserons dans le cours

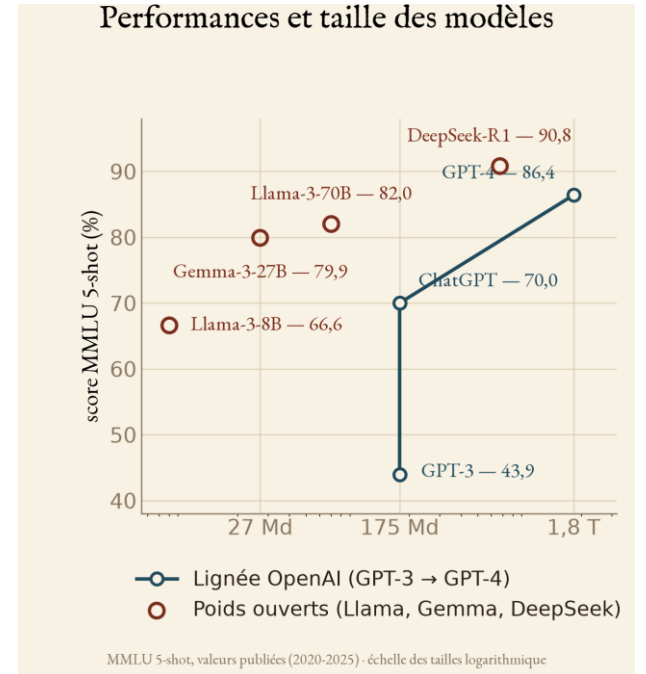
 Qwen Alibaba — la famille ouverte la plus large	 DeepSeek raisonnement ouvert, poids publiés
 Gemma Google DeepMind — du téléphone au poste	 Granite IBM — dense 3, 8 et 30 Md, contexte 512K

Les familles de modèles (2026) — fermés et poids ouverts

Les performances suivent la taille

- GPT-3 (175 Md, 2020) : 44 % MMLU
→ GPT-4 : 86 %
- ChatGPT (2022) : le déclic grand public
— 70 % MMLU
- Ouverts : DeepSeek-R1 à 90,8 % —
harness d'évaluation partagés (lm-evaluation-harness)

#26 · Les modèles de 2026 : noms, tailles, perfs



*MMLU 5-shot — valeurs
publiées par les laboratoires
(2020-2025)*

Des modèles qui pensent

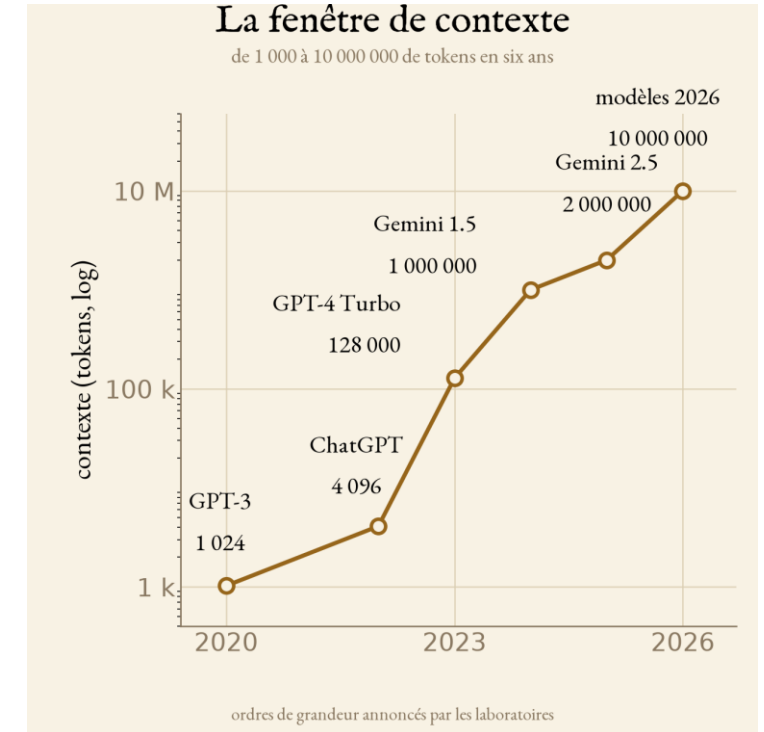
- o1 (2024), DeepSeek-R1 (2025) : le modèle « réfléchit » avant de répondre
- Qwen3 (2025) : mode pensée activable ou non
- Multi-Token Prediction (Meta, 2024) : prédire 4 tokens à la fois, 3× plus vite



Better & Faster Large Language Models via Multi-token Prediction — Gloeckle et al., Meta AI, 2024 (arXiv:2404.19737)

La fenêtre de contexte explose

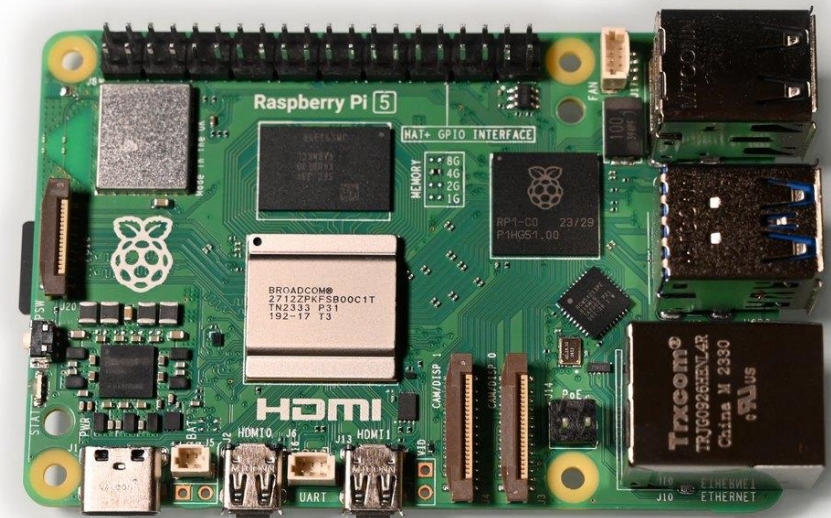
- 2020 : 2 000 tokens (GPT-3)
- 2023 : 200 000 (Claude 2.1) —
2024 : 1 M (Gemini 1.5)
- 2026 : le million de tokens devient standard



*Tokens de contexte (échelle log.)
— x500 entre 2020 et 2024*

Des modèles locaux au niveau frontier

- Gemma 4 E2B/E4B (Google, 2026) : 2-4 Mds effectifs, sur Raspberry Pi 5
- Qwen3.8-27B (Alibaba, 14/08/2026) : open, rivalise avec GPT-5
- Needle 2 (Cactus) : 45 M de paramètres, 14 Mo — jusque sur un ESP32-S3



Raspberry Pi 5 (~80 €) — les modèles locaux tiennent sur la carte

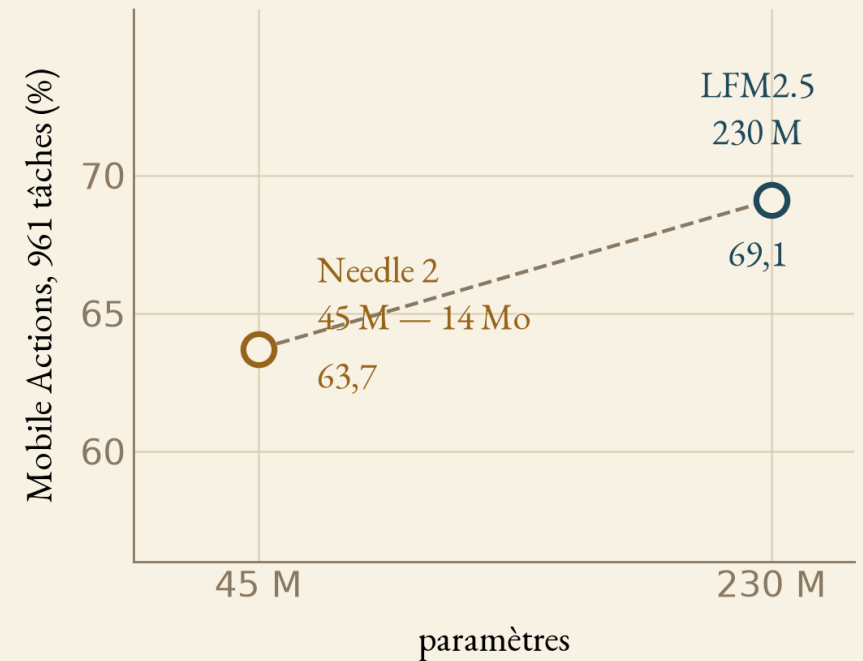
14 MB

Needle 2 (Cactus) : binaire
de 14 Mo, ~28 Mo de
RAM par session

#26 · Les modèles de langage : jokers, tâches, pecks
45 M de paramètres — tool
calling, device use, extraction

Cinq fois plus petit, cinq points de moins

Needle 2 (Cactus) face à un modèle 5× plus gros



évaluation Mobile Actions (961 lignes) publiée par Cactus Compute

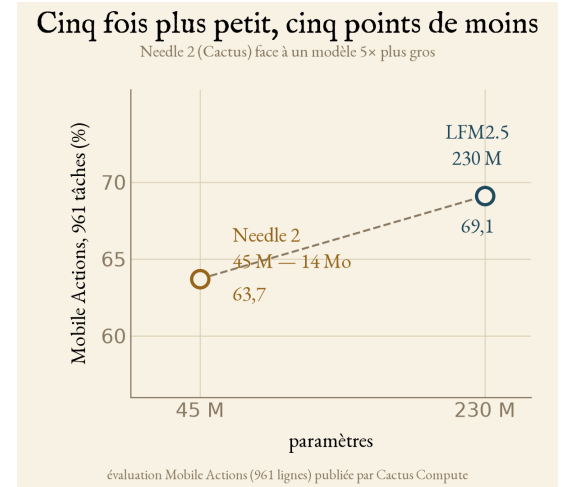
*Needle 2 (45 M, 14 Mo) face à un modèle
5× plus gros — Mobile Actions, 961 tâches
(Cactus Compute)*

Septembre 2026 : le tableau d'ensemble

- Des chatbots aux agents : le modèle devient un système qui agit
- L'IA agentique : outils, mémoire, planification, vérification
- Les prix de l'API ont chuté d'un facteur cent en trois ans
- Les poids ouverts tiennent le niveau de la frontière
- Les modèles locaux montent en qualité — jusqu'au microcontrôleur

Tout petit, tout près : le LLM sur le edge

- Gemma 4 E2B/E4B : 2 à 4 milliards de paramètres, multimodal, sur Raspberry Pi 5
- Needle 2 (Cactus) : 45 M de paramètres, binaire de 14 Mo, tool calling
- Un agent qui tourne sur un microcontrôleur — ESP32-S3, ~28 Mo de RAM
- Le petit modèle agit localement ; le cloud n'est appelé qu'en recours



Needle 2 (45 M, 14 Mo) face à un modèle 5x plus gros — Mobile Actions, 961 tâches (Cactus Compute)

« Les tâches « trop complexes pour être automatisées » deviennent « suffisamment bien décrites pour être déléguées à un agent ». »

— le logiciel devient pilotable par le langage

Conséquence économique

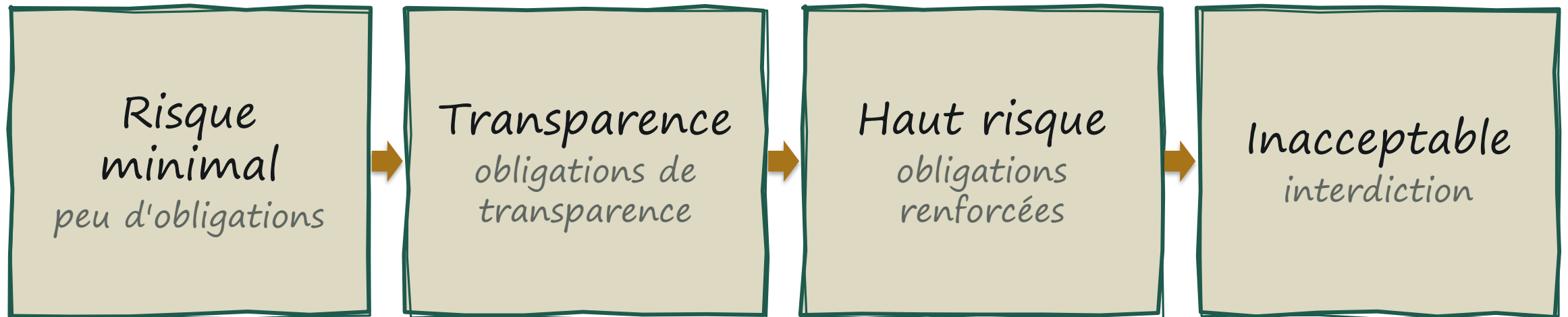
- Compétitivité, automatisation, productivité
- Nouveaux produits, nouvelles interfaces, nouveaux métiers

Le risque change de nature

L'échelle du risque

- Un chatbot qui hallucine : mauvaise réponse
- Un agent qui hallucine : mauvaise action
- Un agent avec accès au mail : fuite potentielle
- Un agent avec accès à un ERP : erreur opérationnelle
- Un agent qui exécute du code : risque de sécurité

L'approche européenne par le risque



2 août 2026

pouvoirs d'exécution de la Commission, amendes
comprises

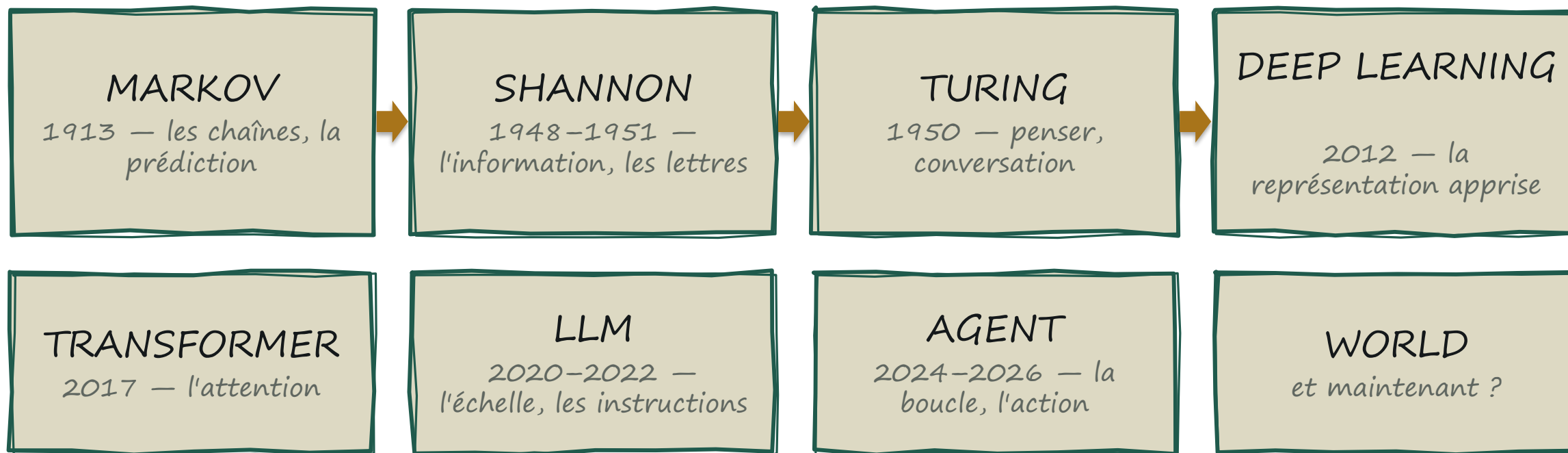
obligations GPAI depuis le 2 août 2025 + transparence (art. 50)

« La question n'est plus « est-ce que l'IA va arriver ? » — elle est déjà là. Comment l'entreprise va-t-elle la gouverner ? »

— le timing est excellent : les règles deviennent opérationnelles

Retour à Markov : 1913 → 2026

La même idée, cent treize ans plus tard : attribuer une probabilité à la suite.



Can a language think?

la réponse est plus intéressante que la question

« Si penser signifie manipuler des représentations, prédire, abstraire, raisonner et produire des actions cohérentes — alors un modèle entraîné à grande échelle sur le langage en réalise une partie substantielle. »

Du mot à l'action

Nous avons commencé par un caractère prédit : où cela s'arrête-t-il ?

