

# IA GÉNÉRATIVES: DES OUTILS POUR LES ÉTUDIANTS, LES ENSEIGNANTS ET LES CHERCHEURS

Lundi 24 mars 2025

IA générative à l'Université Paris-Saclay

Vincent Guigue

<https://vguigue.github.io>





# Les modèles de langue en 5 tableaux

## Modélisation probabiliste de la langue

Découpage des textes = tokens

Large Language Models (LLMs), such as GPT-3 and GPT-4, utilize a process called tokenization. Tokenization involves breaking down text into smaller units, known as tokens, which the model can process and understand. These tokens can range from individual characters to entire words or even larger chunks, depending on the model. For GPT-3 and GPT-4, a Byte Pair Encoding (BPE) tokenizer is used. BPE is a subword tok

Itération du processus

Début de texte

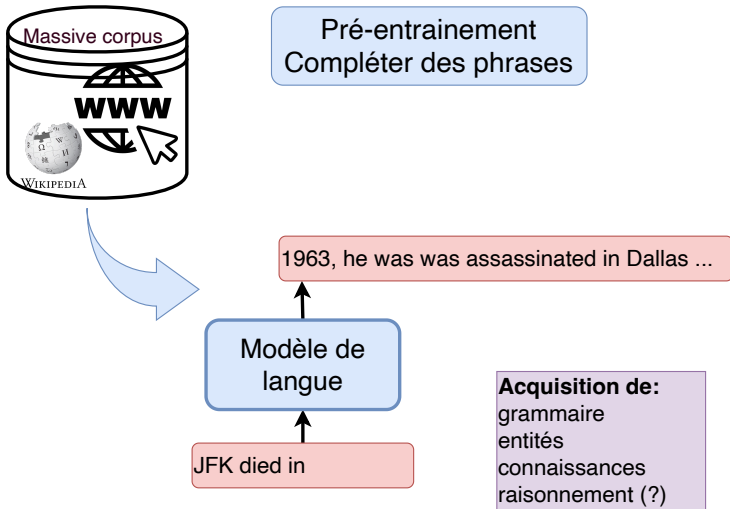
Modèle de langue

Prédiction de la suite

Dictionnaire	Large	0.02
	entire	0.01
	For	0.00
	units	0.00
	...	...
	can	0.00
	may	0.00
...	0.09	
...	...	
...	<b>0.30</b>	



# Les modèles de langue en 5 tableaux





# Les modèles de langue en 5 tableaux

## Instruction finetuning

Please answer the following question.  
What is the boiling point of Nitrogen?

## Chain-of-thought finetuning

Answer the following question by reasoning step-by-step.

The cafeteria had 23 apples. If they used 20 for lunch and bought 6 more, how many apples do they have?



Modèle de langue



Spécialisation sur des tâches

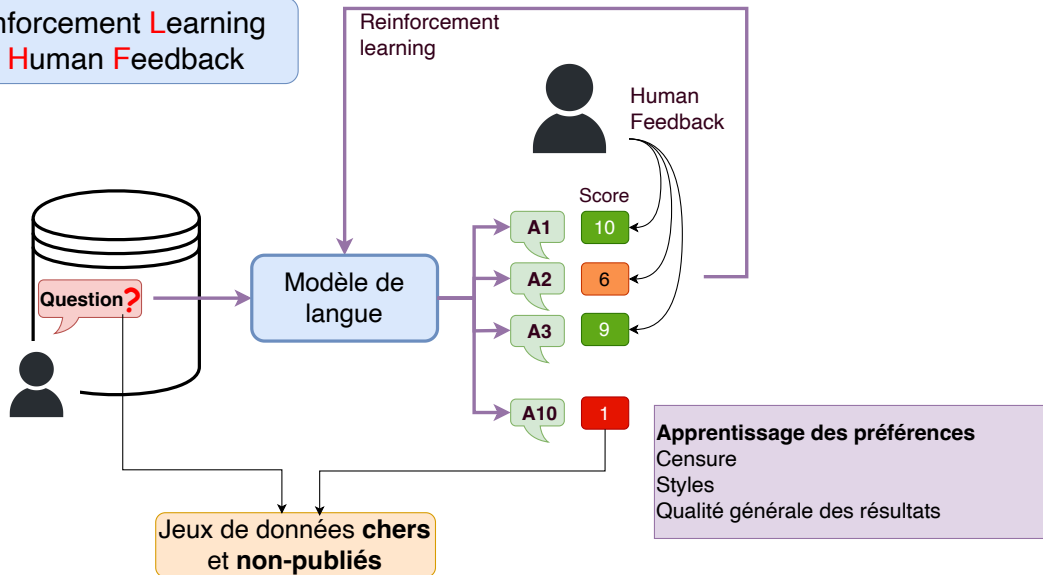
### Acquisition de:

Capacité de répondre à une question  
Suivre un dialogue  
Connaissances physiques  
Bases de raisonnement



# Les modèles de langue en 5 tableaux

Reinforcement Learning  
with Human Feedback





# Chaine des compétences et de la souveraineté

## Construction du modèle de base

### Maitrise des données

- Collecte/équilibrage
- Nettoyage

### Entrainement

- Puissance machine (milliers de GPU)
- Architecture/recherche ML

## Raffinement du modèle

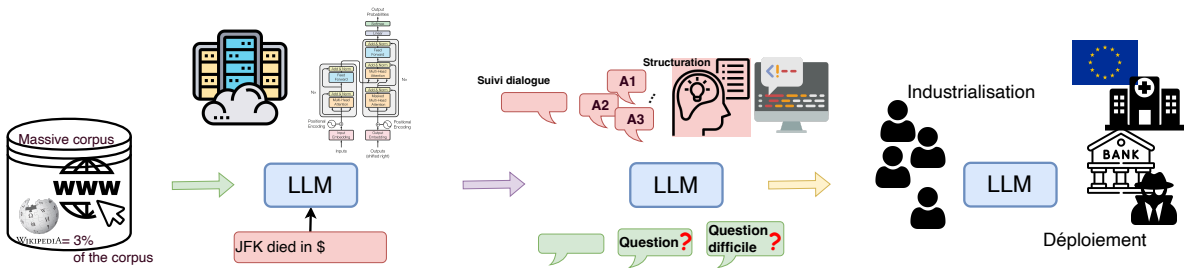
### Maitrise & construction des données

- interactions humaines +++
- prix des données
- spécialisation à la demande

## Exploitation du modèle

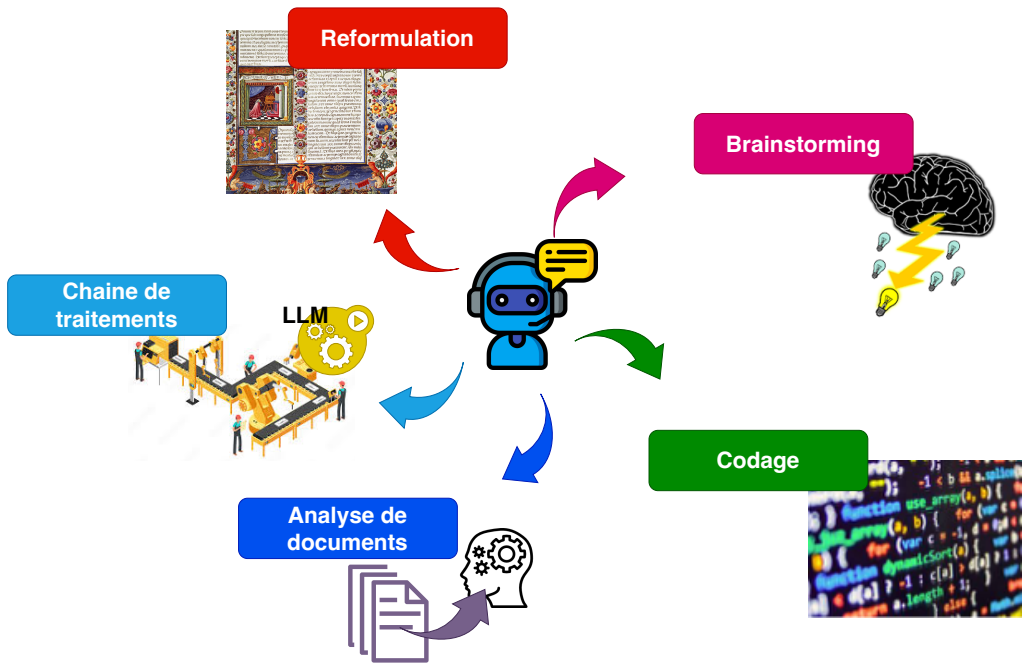
### Optimisation / Limitation du coût

- compétence MLOps
- déploiement local





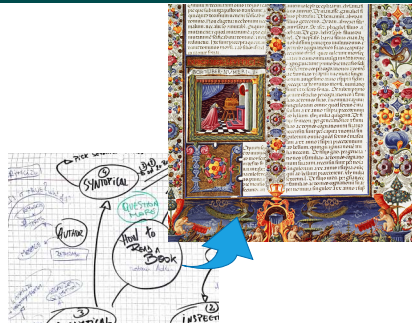
# Les principaux usages en 5 tableaux





# (1) Reformulation, mise en page, traduction

- Lettre de recommandation, motivation etc...
  - Ecriture d'article
  - Traduction
- ⇒ Gain en rapidité, gain en qualité (?)



- Faut-il indiquer l'usage des LLM dans les articles?
  - Quid de la détectabilité?
- Comment traiter les lettres de motivation dans les recrutements?
- Quels comportements sont éthiques ou pas?

[Charte?]

- Mauvaises tournures (et/ou phrases caractéristiques) [cf G. Cabanac]

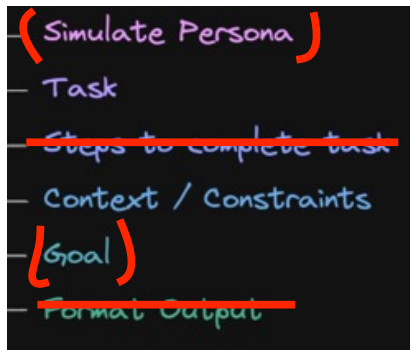
*In the era of... Overlearning...*





## (2) Brainstorming / plan de cours / révision de stats

- Chercher l'inspiration [synd. page blanche]
  - Organiser ses idées rapidement
  - Eviter les oublis
  - Se documenter de manière ciblée, adaptée à ses besoins
- ⇒ Des réponses bluffantes parfois incomplètes ou partiellement fausses... Mais souvent utiles



*3 articles de références sur l'usage des transformers en recommandation*

*A quoi sert la loi poisson log normale?*

*Proposer 10 parties pour un cours sur les Transformers en IA*

- Dans quel périmètre les LLM sont-ils fiables?
- Quels risques pour les sources d'information primaires?
- Quels risques sociétaux sur l'information?



## (3) Codage: différents outils, différents niveaux

- Donner la solution d'un exercice
- Apprendre à coder ou s'y remettre
  - Nouveaux langages, nouvelles approches (ML?)
  - Bénéfier des explications...

Mais comment gérer les erreurs?

- Aide sur une lib [*getting started*]
- Codage + rapide



**GitHub**  
Copilot



- Quid des copyrights?
  - Quel impact sur le traitement futur des codes?
- Comment adapter la pédagogies ?
- Combien d'appels pour la complétion?
  - Quel bilan carbone?
- Quel risque de propagation d'erreurs?

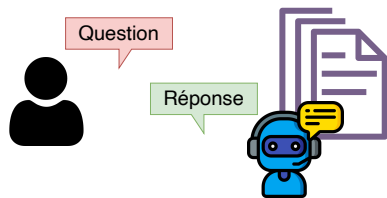
```
serbment.ts  write_sql.go  parse_expenses.py  addresses.rb

1 import datetime
2
3 def parse_expenses(expenses_string):
4     """Parse the list of expenses and return the list of triples (date,
5     Ignore lines starting with #.
6     Parse the date using datetime.
7     Example expenses_string:
8         2016-01-02 -34.01 USD
9         2016-01-03 2.59 DMK
10        2016-01-03 -2.72 EUR
11    """
12    expenses = []
13    for line in expenses_string.splitlines():
14        if line.startswith("#"):
15            continue
16        date, value, currency = line.split(" ")
17        expenses.append((datetime.datetime.strptime(date, "%Y-%m-%d"),
```



## (4) Analyse de documents

- Synthèse de documents / articles
- Dialogue avec une base documentaire
- Assistance à l'écriture de reviews
- FAQ, service d'assistance interne dans les entreprises
- Veille
- Génération de quizz à partir d'un poly de cours



NotebookLM

Think **Smarter**,  
Not Harder

Try NotebookLM

- Les articles seront-ils encore lus demain?
  - Faut-il rendre nos articles NotebookLM-proof?
- Comment gagner du temps mais rester honnête & éthique?



# (5) LLM dans une chaine de production / Agentic AI

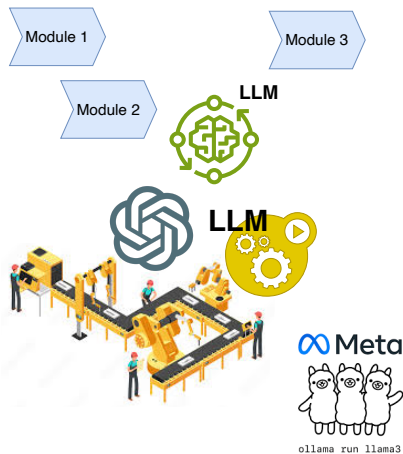
- Extraire des connaissances
- Trier des documents / produire résumés
- Générer des exemples pour entrainer un modèle  
[Teacher/student - distillation]
- Variantes d'exemples ↗ ↗ taille données  
[Data augmentation]

⇒ Intégrer le LLM dans une chaine de traitement  
= peu/moins de supervision = **Agentic AI**



format de sortie ⇒ regex / contrainte de formattage JSON

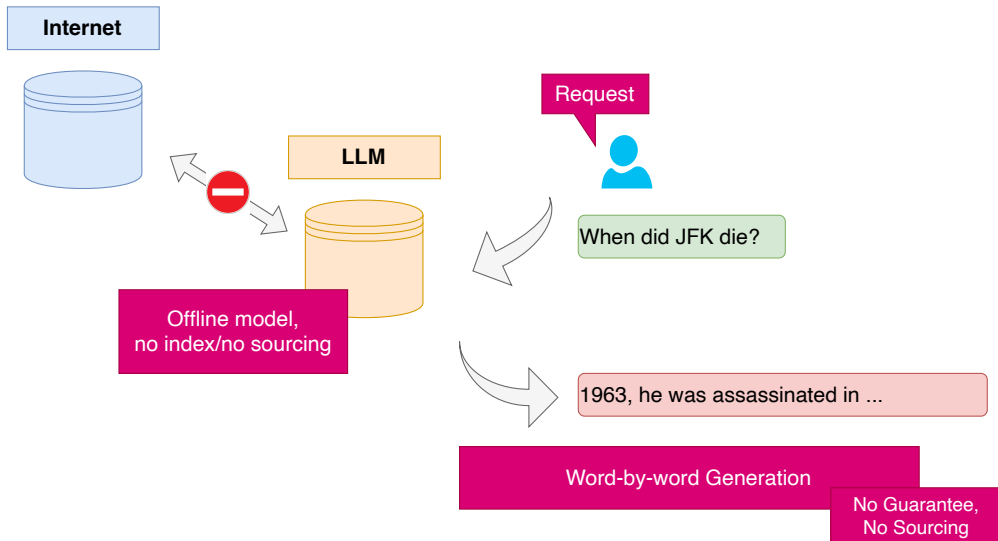
- Est ce que je peux entrainer des modèles sur des données générées?
- Combien ça coute? (\$ + CO<sub>2</sub>) Besoin de GPU?
- Que valent les modèles open-weight?





# Mémoire paramétrique vs extraction d'information

Besoins spécifiques: métriques, enjeux...

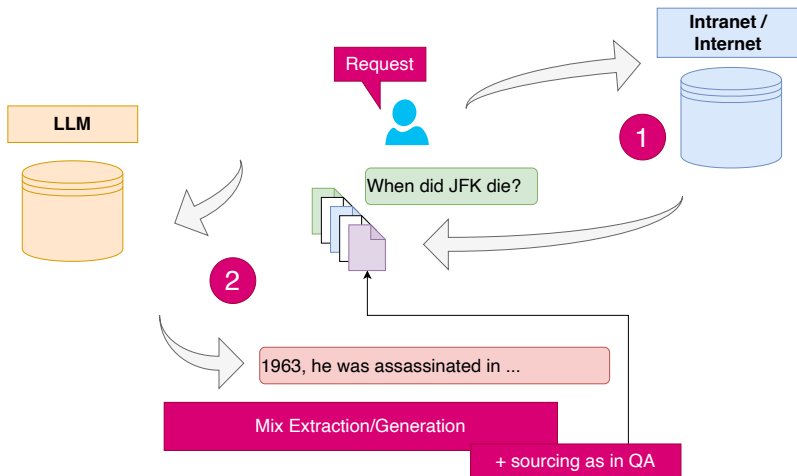




# Mémoire paramétrique vs extraction d'information

Besoins spécifiques: métriques, enjeux...

[≠ hallucinations !!]



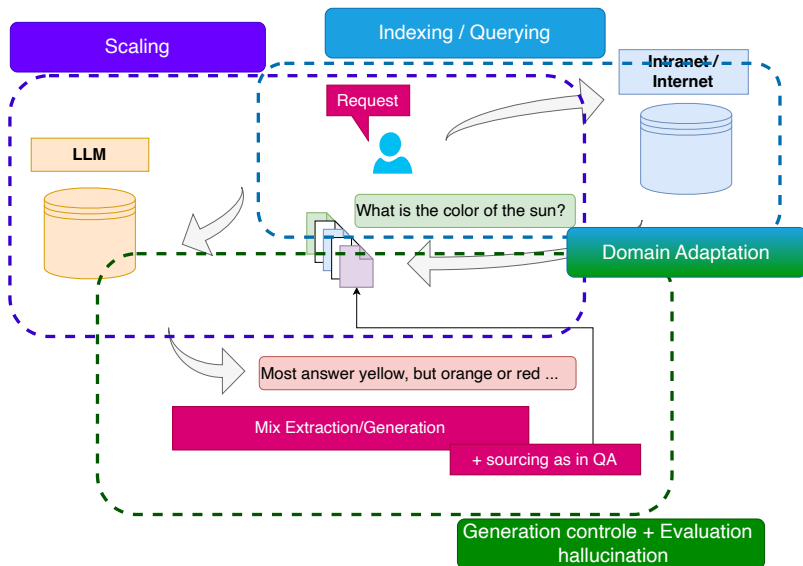
NotebookLM : penser qu'un ensemble de documents (longs) peuvent constituer une source pour le dialogue



# Mémoire paramétrique vs extraction d'information

Besoins spécifiques: métriques, enjeux...

[≠ hallucinations !!]

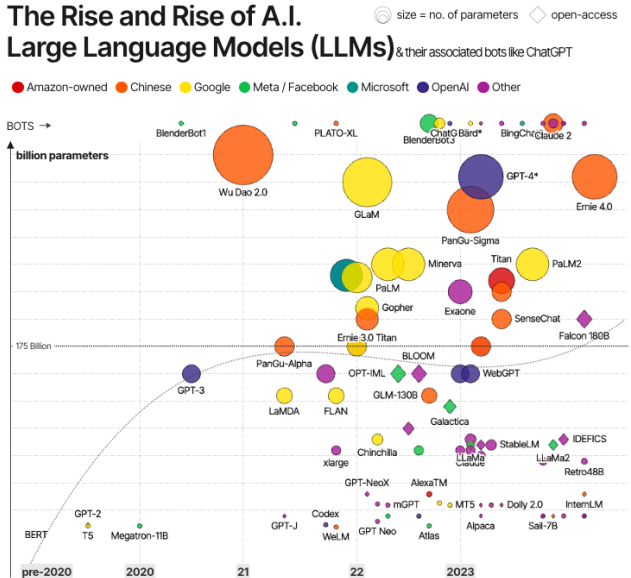




# Les LLMs & la frugalité

## The Rise and Rise of A.I.

### Large Language Models (LLMs) & their associated bots like ChatGPT



David McCandless, Tom Evans, Paul Barton  
Information is Beautiful // UPDATED 2nd Nov 23

source: news reports, [LifeArchitect.ai](https://life-ai.com)  
\* = parameters undisclosed // see [the data](#)

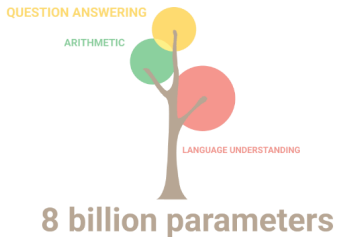
## # Paramètres

- 1998 LeNet-5 = 0.06M
- 2011 Senna = 7.3M
- 2012 AlexNet = 60M
- 2017 Transformer = 65M / 210M
- 2018 ELMo = 94M
- 2018 BERT = 110M / 340M
- 2019 GPT2 = 1,500M
- 2020 GPT3 = 175,000M



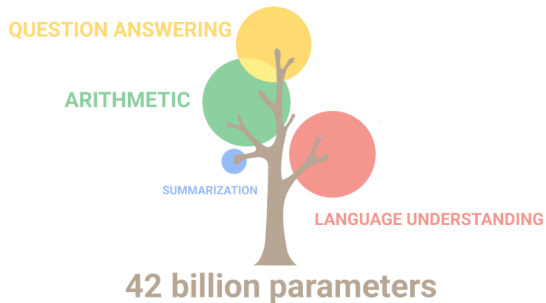


## Capacités émergentes



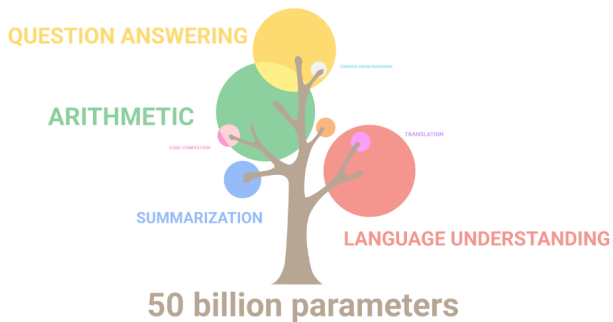


## Capacités émergentes



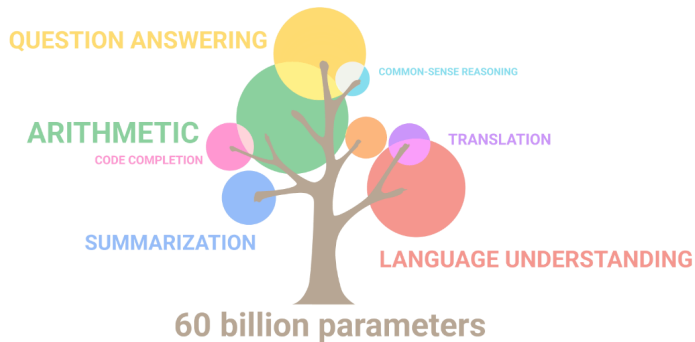


## Capacités émergentes



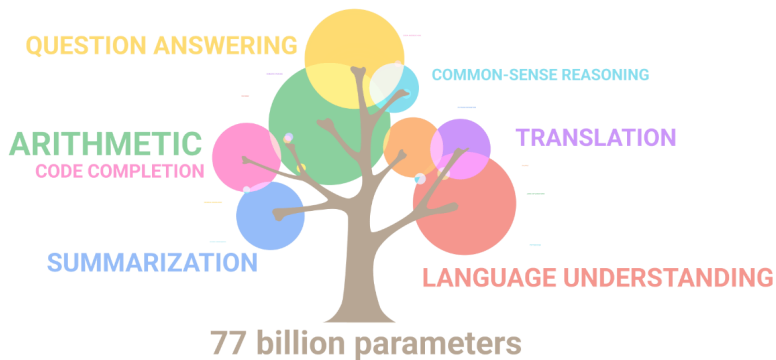


## Capacités émergentes



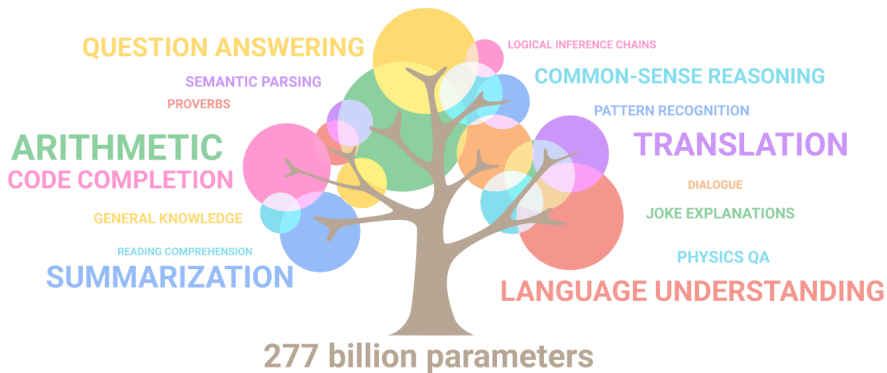


## Capacités émergentes



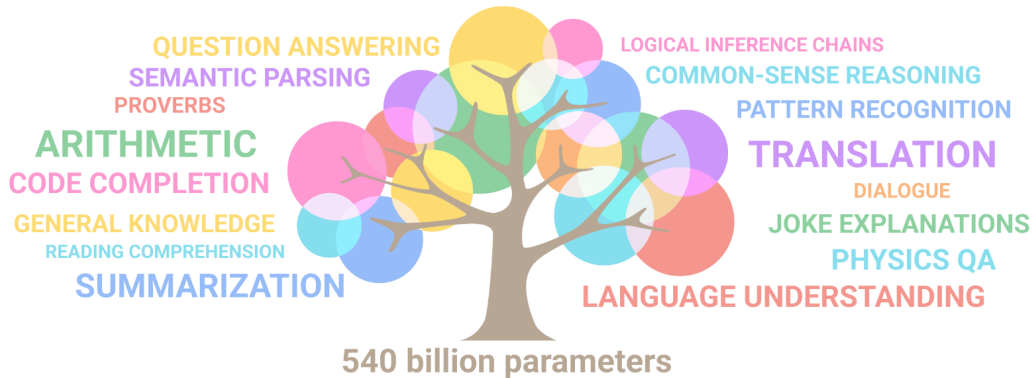


## Capacités émergentes



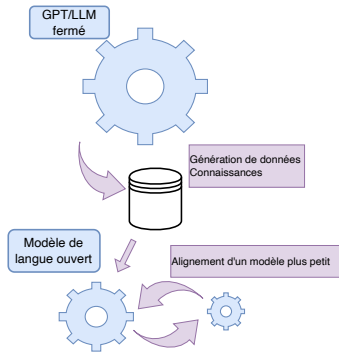


## Capacités émergentes





## Distillation



## Elagage Quantification

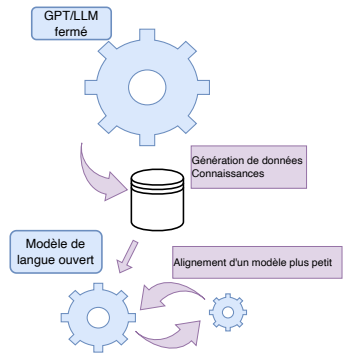
## Mixture of experts

La frugalité... Taille modèle **x1000 en 3 ans...** Puis **opti. x1/100 en 2ans**

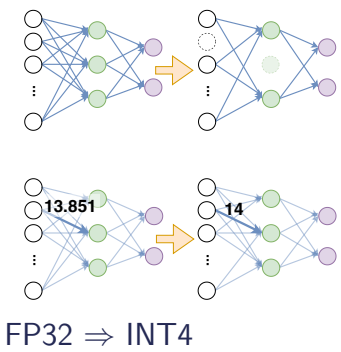




## Distillation



## Elagage Quantification

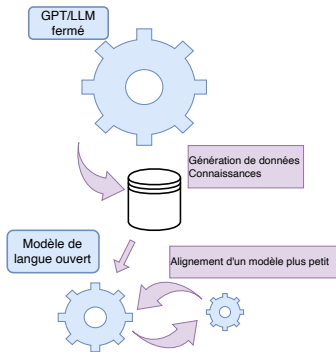


## Mixture of experts

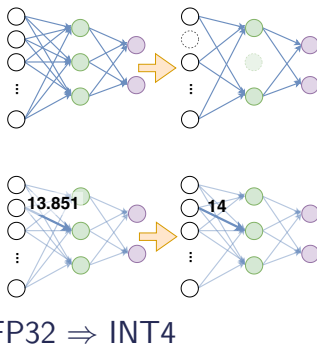
La frugalité... Taille modèle **x1000 en 3 ans...** Puis **opti. x1/100 en 2ans**



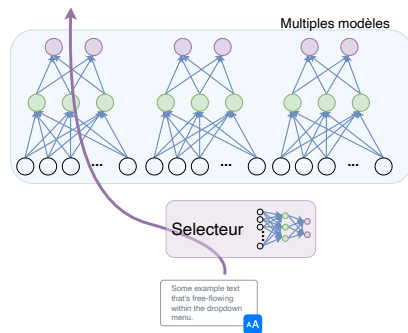
## Distillation



## Elagage Quantification



## Mixture of experts



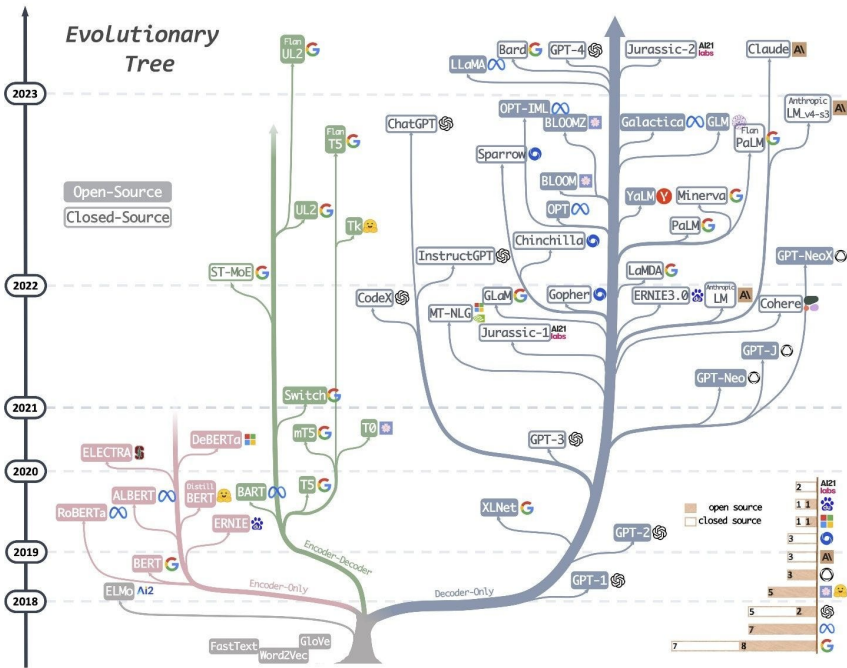
+ Industrialisation du code

Mais le plus important: adéquation aux usages!

La frugalité... Taille modèle **x1000 en 3 ans...** Puis **opti. x1/100 en 2ans**

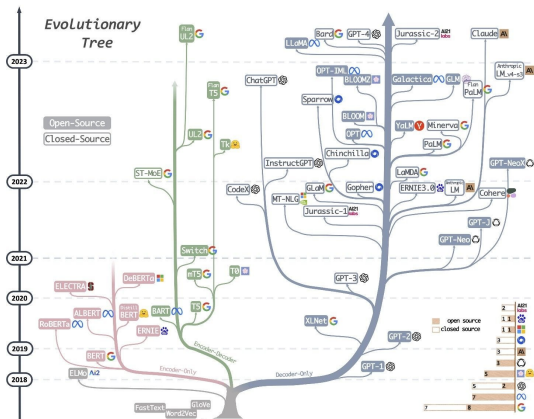


# Les LLMs, la transparence & la privacy





# Les LLMs, la transparence & la privacy





## Transparence

- Est ce que je peux le modifier? **Adaptation**
- Quelles données d'entrainement? **Data contamination / skills**
- Quelle ligne éditoriale / censure? **Accès à l'information**
- Pourquoi cette réponse? **Explicabilité / interprétabilité**

## Privacy

- Fonctionnement local **A quel coût?**
- Quelles garanties sur les serveurs? **Contrats ⇔ US Patriot Act**

## Des solutions disponibles:

- Clé en main: Ollama 
  - Nombreux modèles disponibles + MAJ
  - Accès API / console
- Sur mesure : HuggingFace 
  - Pytorch/TensorFlow
  - Accès modèles + fine-tuning
  - Quantif / optim



Open-Weight  $\neq$  Open-Source

## Transparence

- Est ce que je peux le modifier?  
Adaptation
- Quelles données d'entrainement?  
Data contamination / skills
- Quelle ligne éditoriale / censure?  
Accès à l'information
- Pourquoi cette réponse?  
Explicabilité / interprétabilité

## Privacy

- Fonctionnement local  
A quel coût?
- Quelles garanties sur les serveurs?  
Contrats  $\Leftrightarrow$  US Patriot Act

## Des solutions disponibles:



- Clé en main: Ollama
  - Nombreux modèles disponibles + MAJ
  - Accès facile
- Sur mesure
  - Python
  - Accès
  - Qualité

### DeepSeek

- Toujours des Transformers / 600 Milliards de paramètres
- Mixture of experts (comme Mixtral)
- 2000 GPU NVidia + 5M \$/training (=chatGPT)
- ... Un critère d'apprentissage original
  - + performant pour le raisonnement



Open-Weight  $\neq$  Open-Source

## Transparence

- Est ce que je peux le modifier?  
Adaptation
- Quelles données d'entraînement?  
Data contamination / skills
- Quelle ligne éditoriale / censure?  
Information
- ... quel coût?  
abilité
- Quelles garanties sur les serveurs?  
Contrats  $\Leftrightarrow$  US Patriot Act



## ■ **Quid des hallucinations?**

- Faut-il les réduire ou vivre avec?
- Les LLM vont-ils progresser? Dans quelles directions?
- Le LLM nous fait-il *perdre* le rapport à la vérité? à la vérification?

## ■ **Faut-il des petits ou des grands modèles de langues?**

- Combien ça coute? Est-ce soutenable?
- Avec ou sans fine-tuning?
- Qu'est ce que la frugalité dans l'univers des LLM?

## ■ **Quand les autres s'en servent... Quel impact sur moi?**

- Productivité (collègues chercheurs, codeurs, relecteurs, ...)
- Pédagogie : gérer/former des étudiants *branchés*

## ■ **Protection des données... Les miennes et celles des autres**

- Est-il raisonnable d'entraîner les LLM sur github, wikipedia, les articles scientifiques, les journaux, ... ?
- Quelle est l'importance de la privacy? Quels risques lorsque j'utilise un LLM?



## ■ Quid des hallucinations?

- Faut-il les réduire ou vivre avec?
- Les LLM vont-ils progresser? Dans quelles directions?
- Le LLM nous fait-il *perdre* le rapport à la vérité? à la vérification?

## ■ Faut-il des petits ou des grands modèles de langues?

- Combien ça coûte? Est-ce soutenable?

- A Le smartphone a fait de moi un *humain-augmenté*...

- Q Est ce que le LLM va faire de moi un *chercheur-augmenté*?

## ■ Quant

⇒ Jetez (quand même) un oeil à NotebookLM

- Productivité (collègues chercheurs, codeurs, relecteurs, ...)
- Pédagogie : gérer/former des étudiants *branchés*

## ■ Protection des données... Les miennes et celles des autres

- Est-il raisonnable d'entraîner les LLM sur github, wikipedia, les articles scientifiques, les journaux, ... ?
- Quelle est l'importance de la privacy? Quels risques lorsque j'utilise un LLM?