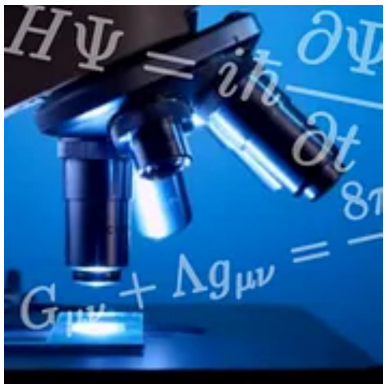
http://pierrealainmillet.fr/ChatGPT-va-t-il-chercher-ses-reponses-dans-sa-base-de

ChatGPT va-t-il chercher ses réponses dans sa base de données ?

- Numérique -



Date de mise en ligne : jeudi 11 décembre 2025

Copyright © Blog Vénissian de Pierre-Alain Millet - Tous droits réservés

Je vous propose une lecture scientifique qui me parait très pédagogique sur un sujet d'actualitéâ€¦ l'IA et ces outils de dialogue qui nous surprennent.. Cela fait un moment que j'en utilise pour aller plus vite dans l'écriture de mes interventions et que je suis toujours déçu car cela ne fait jamais ce que j'ai enviesâ€¦ sauf pour corriger l'orthographe, mettre au propre une transcription orale, ou faire un résumé de texteâ€¦

Je pense que vous êtes nombreux à essayer chatgpt, et peut-être Mistral que je vous conseille car il est français, ou même deepseek, l'outil chinois qui a un gros avantage aussi, il est en « open source » donc on sait exactement ce qu'il fait..

Et je sais aussi que pour la plupart d'entre nous, on se demande bien comment ça marche et qu'il y a beaucoup d'incompréhensionâ€¦ dont celle qu'évoque cet article. Est-ce que chatgpt est une incroyablement énorme base de donnée qui sait tout sur tout et qui peut donc répondre à toutes nos questionsâ€¦ ?

Et bien non, chatgpt, comme tous les autres outils d'IA ne fait que nous proposer des réponses « probables », sauf qu'il a beaucoup apprisâ€¦

Cet article d'un scientifique que j'aime bien écouter sur youtube ou lire sur sa chaine substack m'a paru utile à vous faire connaître, car il me semble être très pédagogique, et éclaire un sujet très important aujourd'hui pour apprendre à utiliser ces outils sans croire qu'ils seraient « magique »â€¦

pam

Trois ans après sa sortie, retour sur cette incompréhension persistante concernant le fonctionnement de ChatGPT et consorts. Et pourquoi c'est à la fois la force et la faiblesse de ces modèles.

Science étonnante

déc. 11, 2025

[http://pierrealainmillet.fr/local/cache-vignettes/L400xH210/10000000000005b0000002fe47a6b74c-3b5f1.jpg]

Ce sont des affirmations que j’entends encore trop souvent :

- « ChatGPT va chercher dans ses données »,
- « Il va piocher parmi les sites qu’il a en mémoire »,
- « Si ChatGPT a un truc faux dans sa base de données, il va le ressortir »,
- « Il ne fait que recopier les trucs qu’il a déjà lu ». Si vous n’utilisez que peu ces modèles, je conçois que cela

puisse sembler plausible. Et pourtant, ça n'est vraiment pas comme ça que ChatGPT (ou ses cousins) fonctionnent !

Les chatbots modernes sont basés sur des grands modèles de langage (les LLMs en anglais). Et ces modèles sont effectivement fabriqués en ingurgitant une quantité astronomique de données textuelles (« tout internet », pour le dire vite). Mais pour développer leurs capacités, ils ne sont pas programmés explicitement avec une base de données. Ils sont plutôt entrainés à réaliser leur tâche : on parle de machine learning.

Et pour comprendre pourquoi il n'existe pas de « base de données » derrière ces modèles quand on les utilise, prenons exemple sur ce qui est la forme la plus élémentaire de machine learning : la régression linéaire.

Un peu de botanique fictive

Imaginez que vous soyez un spécialiste de la croissance des arbres. Vous étudiez une certaine espèce, et pour cela vous allez collecter des données dans un coin de forêt.

Pour une quarantaine d'arbres, vous mesurez à la fois le diamètre et la hauteur. Une fois revenu au labo, vous rentrez tout ça dans Excel.

[http://pierrealainmillet.fr/local/cache-vignettes/L284xH372/100000000000022c000002d8dba6fa54-f973d.png]

Une fois là, vous décidez alors de tracer les données sur un graphique, et voici ce que vous observez.

[http://pierrealainmillet.fr/local/cache-vignettes/L400xH300/10000000000005b00000044489270cd2-600aa.png]

C'est intéressant ! Il semble y avoir une belle relation entre la hauteur d'un arbre et son diamètre.

Une façon habituelle de la faire apparaître, c'est de réaliser une régression linéaire.

[http://pierrealainmillet.fr/local/cache-vignettes/L400xH300/10000000000005b0000004447b43197c-a460e.jpg]

Je vous ai affiché la formule qui résulte de la régression : dans ce cas précis, notre régression nous dit que la hauteur d'un arbre est environ 0.09 fois son diamètre, auquel on ajoute 1.06m. Simple !

Faire des prédictions

Une fois en possession de cette régression linéaire, j'ai quelque chose de formidable : une machine à prédire la hauteur des arbres ! Eh bien oui, c'est fastidieux de mesurer la hauteur d'un arbre. Alors maintenant si vous me donnez juste son diamètre, je peux le rentrer dans ma régression et vous faire une prédiction pour sa hauteur. Il me suffit de multiplier le diamètre par 0.09 et d'ajouter 1.06.

Et notez bien que pour faire cette simple opération, je n'ai plus besoin des données d'origine ! Mon chien peut bien avoir mangé la feuille où j'avais noté mes observations, ma régression linéaire est toujours là.

Pour symboliser cette idée, je vais à moitié effacer les observations pour qu'on se souvienne qu'on n'y a plus accès.

Copyright © Blog Vénissian de Pierre-Alain Millet

[http://pierrealainmillet.fr/local/cache-vignettes/L400xH300/10000000000005b000000444e6aed619-e5b00.jpg]

Grâce à ma régression linéaire, si vous me montrez un arbre de la même espèce, et dont le diamètre fait environ 100cm, je peux estimer qu'il fera approximativement 10m de haut. J'ai juste appliqué ma formule.

Si vous ne m'avez pas vu faire cette régression, au moment où je vous donne cette estimation, vous pourriez être tentés de dire « Ah mais tu as juste été chercher dans ta base de données un arbre similaire et tu as recopié la valeur ». Mais non ! J'ai vraiment juste utilisé la régression dont j'ai appliqué la formule. Je n'ai plus les données, elles ont été jetées et oubliées depuis.

Une régression, c'est un résumé des données, une sorte de compression des relations statistiques qui existent entre elles.

L'entraînement est une compression

Je parle de compression, car là où ma base initiale contenait 40 arbres (donc 80 observations, c'est-à-dire 80 nombres), ici je n'ai plus que DEUX nombres : la pente (0.09) et l'ordonnée à l'origine (1.06) de ma régression. Ces deux nombres sont une sorte de synthèse des 80 observations initiales. Ils n'en capturent pas exactement toute les nuances, mais ils en résument les grandes régularités statistiques : ils sont une compression du savoir contenu dans la base de données initiale.

C'est la même chose pour les LLMs, mais à une échelle bien plus importante. Pour fixer les idées, un grand modèle de langage aujourd'hui, c'est typiquement quelques dizaines à quelques centaines de milliards de paramètres (on les appelle parfois « les poids » du modèle). Ces milliards de paramètres sont les équivalents des deux coefficients de ma régression linéaire : ce sont eux qui vont être ajustés pendant l'entrainement, et qui vont contenir à la fin les relations statistiques ainsi découvertes.

Pour déterminer les paramètres de ces modèles énormes, il est nécessaire de les entrainer sur des dizaines de milliers de milliards de mots, c'est-à-dire "tout internet". (Notez qu'on retrouve un peu le même genre de ratio que dans mon exemple : le taille de l'ensemble d'entrainement en nombre de tokens est typiquement de quelques dizaines à quelques centaines de fois le nombre de paramètres.)

Cette idée d'entrainement sur des données explique aussi pourquoi les LLMs ont une limite temporelle à leurs connaissances. Ils ont été entrainés sur un certain corpus de textes, qui a été figé à un certain moment, et ils ne peuvent donc pas avoir connaissance d'événements postérieurs à leur date de construction. On voit encore trop de gens qui rigolent bêtement de voir qu'un LLM ignore le résultat du match de la veille ou le changement de gouvernement d'il y a 3 semaines.

Robustesse, hallucinations … et créativité !

Essayons de comprendre les avantages et les inconvénients d'un LLM par rapport à un hypothétique algorithme qui irait « piocher dans sa base de données de textes. » Et pour cela, revenons sur mon exemple des arbres.

Si vous me demandez : « quelle est la hauteur d'un arbre qui fait 84cm de diamètre ? », ma régression va répondre « environ 8 mètres ». J'ai à nouveau juste appliqué la formule. Et pourtant regardez les données de départ dont je

disposais : le seul arbre de ce diamètre faisait plutôt 9.5 m.

[http://pierrealainmillet.fr/local/cache-vignettes/L400xH300/10000000000005b000000444d1e4d7da-60d46.jpg]

Mais je le répète : ma régression n'a pas accès aux données de départ ! Elle n'a aucune idée qu'il existait un arbre de 84cm de diamètre et 9.5m de hauteur. Ca n'est pas une base de données, c'est une compression statistique. Et cela peut être un avantage…ou un inconvénient !

Ici l'arbre de 9.5m ressemble à une anomalie statistique : peut-être était-il particulier, ou bien a-t-il été mal mesuré. Ma régression linéaire est donc plus robuste que si j'allais simplement chercher un exemple similaire dans les données. Si j'avais recopié cette donnée de la base, ma réponse aurait probablement été moins juste. La régression linéaire a permis d'atténuer l'impact de ce point qui semble un peu anormal.

C'est la même chose avec les LLMs : s'ils lisent un truc faux ou absurde pendant leur entrainement, cela ne se reflètera pas forcément dans ce qu'ils produiront ensuite. Ils ont une certaine robustesse conférée par la procédure d'entrainement : un site web faux ne suffit pas à faire dire n'importe quoi à un LLM sur le sujet concerné.

Par rapport à une simple "base de données", l'autre intérêt évident des modèles de machine learning en général (et des LLMs en particulier), c'est qu'on peut les utiliser dans une certaine mesure dans des domaines qui sortent un peu de leurs données d'entrainement. C'est évidemment quelque chose d'impossible si on va juste "regarder la base de données".

Si vous me montrez un arbre de diamètre 20cm ou 160cm, je n'ai rien qui corresponde dans ma base. Ce sont des valeurs extrêmes et je n'ai pas trouvé d'arbre de ce genre. Mais avec ma régression, je pourrais faire une estimation de leur hauteur qui ne serait probablement pas trop mauvaise. Je fais de l'extrapolation.

Mais il ne faut pas pousser ces extrapolations trop loin de leur zone d'apprentissage : si j'applique naïvement ma régression, elle me dit par exemple qu'un arbre de diamètre nul ferait environ 1m de hauteur…c'est certainement faux ! Quand on extrapole trop loin, on dit qu'on "sort de la distribution" d'entrainement, et c'est cette utilisation "hors de la distribution" qui peut causer les hallucinations des LLMs.

[http://pierrealainmillet.fr/local/cache-vignettes/L400xH299/10000000000005b000000444e6aed619-2-a5ab0.jpg]

Mais les utilisations qui sont juste légèrement hors distribution sont au contraire sources de créativité. Regardez à nouveau mon graphique ci-dessus. Si vous me montrez un arbre d'un diamètre de 70cm, je peux vous donner une très bonne estimation de sa hauteur. Pourtant dans mes données, il n'y a aucun arbre de cette taille : il y a un trou entre 60 et 80cm. Mais ça n'est pas grave, si les relations statistiques découvertes par le modèle sont robustes, elles ont une capacité d'interpolation. On peut les utiliser dans des zones non explorées mais qui combinent des zones que l'on connait.

Dans le cas des modèles de langage (et c'est vrai aussi des modèles génératifs pour les images et les vidéos), ces capacités d'interpolation peuvent devenir de véritables capacités créatives ! Ca nous parait difficile à concevoir car ces modèles ont une dimension incroyablement plus élevée que les régressions linéaires, et il est difficile de s'imaginer ce que veut dire « interpoler » dans un espace à plusieurs milliards de dimensions. Et pourtant c'est bien de ça dont il s'agit quand un LLM est capable de suggérer des pistes créatives pour votre prochain scénario, ou de démontrer un résultat nouveau en mathématiques.

Pour ma part, je crois qu'on tient en un peu trop haute estime la créativité humaine, et qu'on la place un peu trop sur un piédestal comme quelque chose d'incroyable, d'unique et de presque magique. Or mon expérience personnelle (c'est-à-dire les fois où j'ai été considéré comme « créatif » dans des domaines comme la vulgarisation, la recherche

Copyright © Blog Vénissian de Pierre-Alain Millet

scientifique, la musique ou le game design), ce qu'on appelle "créativité" se résume bien souvent à astucieusement combiner au bon moment des choses que personne n'avait pensé à associer comme ça avant. Ca n'enlève rien au mérite et à la valeur de ces actes créatifs, mais reconnaissons que ça n'est pas non plus de la magie !

Or faire ça â€" combiner des choses un peu éloignées mais qui ensemble apportent une réponse intéressante et utile â€" c'est typiquement le genre de chose que les LLMs font très bien par interpolation !

Bien sûr, si vous emmenez un LLM (où n'importe quel autre modèle génératif de ce genre) dans une zone trop éloigné de son ensemble d'entrainement, il va se mettre à défaillir. Et pourtant, cela ne limite en rien leur utilité, tant il y a de place pour faire des interpolations dans des zones où les modèles répondent parfaitement. Et ce, même s'ils n'ont jamais vu de données exactement identiques (comme dans mon exemple de l'arbre de 70cm.)

Et pourtant une fois il a recopié !

Alors ok, les modèles n'ont pas de base de données. Et pourtant on a tous vu des cas où ils recrachent sans sourciller le contenu exact d'un site (par exemple un article Wikipédia), ou bien recopient quasiment une certaine image de leur ensemble d'entrainement. On peut comprendre ce phénomène.

Regardez ces nouvelles données (toujours fictives) pour mon exemple des arbres.

[http://pierrealainmillet.fr/local/cache-vignettes/L400xH250/10000000000005b00000038ed4cf946f-c1d39.jpg]

J'ai fait aussi une régression linéaire, et elle semble tout à fait valide. Mais regardez bien, il se passe un truc nouveau dans ces données : autour d'un diamètre de 40cm, tous les arbres ont une hauteur très bien définie, la variabilité est beaucoup plus faible (ce sont des données fictives, mais admettons qu'il y ait peut-être une raison à cela). Cela ne change pas l'estimation que fait la régression linéaire, mais ça change l'utilisation que certains modèles de machine learning peuvent faire de ces données.

Cela ne se voit pas sur une simple régression linéaire, mais cette variabilité est très importante pour les modèles génératifs comme les LLMs. En effet les modèles génératifs en général ne se contentent pas de prédire une valeur (par exemple la hauteur d'un arbre), mais ils essayent d'en estimer la distribution de probabilité (donc la variabilité autour des valeurs probables).

Ici un modèle plus avancé qu'une régression linéaire classique, et capable d'estimer la variabilité locale, produirait quelque chose de ce genre. La zone en rouge désigne un intervalle de confiance à 95%.

[http://pierrealainmillet.fr/local/cache-vignettes/L400xH250/10000000000005b00000038e4084aa30-7d2ba.jpg]

Imaginons que vous utilisiez le modèle derrière cette figure comme modèle génératif pour créer des données fictives d'arbres. Pour un diamètre de 100cm, le modèle vous proposera aléatoirement des hauteurs entre 9 et 11m (l'intervalle de confiance à 95%, ça veut dire que 95% des arbres générés seront dans la bande rouge). Mais vous voyez que pour un diamètre de 40cm, il ne pourra renvoyer quasiment que la hauteur de 5m. La relation statistique dans cette zone est si forte que cela empêchera le modèle génératif de proposer autre chose qu'un arbre de 5m. A cet endroit, la relation statistique a quasiment mémorisé par cœur le lien entre diamètre et hauteur.

Eh bien dans les LLMs, c'est pareil, et cela dépend aussi de la variabilité des données d'entrainement. Si un LLM essaye de compléter le mot « John », il considèrera plein d'options pour mettre un nom de famille derrière. Car derrière « John », dans son ensemble d'entrainement, il a déjà vu plein de choses différentes.

Copyright © Blog Vénissian de Pierre-Alain Millet

Maintenant si un LLM voit le prénom « Barack »â€¦il y a de très bonnes chances qu'il soit tenté à 99% de le faire suivre par « Obama », car cette relation statistique est extrêmement forte dans les données d'entrainement. Tout se passe comme si le LLM avait « mémorisé » par cœur l'exemple « Barack Obama » de sa base de données. Mais comprenez bien que c'est une mémorisation statistique : il ne va piocher nulle part cette information en particulier. Il n'a plus d'accès direct aux milliers de phrases qu'il a lu et qui parlaient de Barack Obama. Il a juste encodé la relation statistique avec une très forte probabilité. Et c'est cette idée qui peut laisser penser que les LLMs "recrachent par cœur" des sites qu'ils ont lu.

(PS : Pour ceux qui connaissent, je parle évidemment là des modèles de fondation, ceux qui sont juste pré-entrainés à prédire le token suivant, et donc avant le fine-tuning pour en faire un chatbot, qui peut évidemment modifier ces comportements.)

Le cas des outils

Pour être tout à fait complet sur cette question de la "base de donnée" de ChatGPT, je voudrais dire un mot des outils dont disposent les LLMs modernes, et notamment leur capacité à aller chercher sur internet.

On l'a dit, au départ, un LLM est juste une machine à prédire les prochains tokens. Il n'a pas de capacité à « faire autre chose », comme aller chercher dans une base de données. Et pourtant parmi les améliorations qui ont été ajoutées à ces modèles, on trouve la possibilité de faire appels à des « outils ».

En gros c'est comme à « Qui veut gagner des millions ? », chaque fois qu'un chatbot reçoit une demande, il a la possibilité soit de répondre directement à l'utilisateur, soit de demander l'aide d'un outil. Mais comme un LLM ne sait faire rien d'autre que produire du texte, cette demande d'aide se fait au moyen de tokens particuliers qui ne vous seront pas montrés, mais qui seront interceptés et traités par le système mis en place par le fournisseur du LLM.

Prenons un exemple. Si vous demandez à un LLM, la date de la bataille de Hastings, il pourra peut-être produire directement des tokens comme « La bataille de Hastings a eu lieuâ€¦ », et dans ce cas ces tokens vous seront directement destinés à vous, l'utilisateur : ils s'afficheront sur votre écran. Mais il se peut qu'il produise au lieu de cela des tokens spéciaux comme « [tool = RechercheWeb("Hastings")] ».

Dans ce cas, ces tokens ne vous seront pas affichés ! Ils seront interceptés par un code intermédiaire (qui joue comme un rôle de routeur) et qui déclenchera une recherche internet (effectuée par un logiciel classique, pas un LLM). Le résultat de la recherche sera alors retourné au LLM, qui, armé de ces informations, pourra alors vous produire une réponse qui vous sera cette fois destinée.

[http://pierrealainmillet.fr/local/cache-vignettes/L400xH225/10000000000005b000000333a5451ff0-3e524.jpg]

A gauche, quand un LLM vous répond directement. A droite, quand il émet des tokens spéciaux pour appeler un outil (comme une recherche web), outil qui sera alors traité par un système externe au LLM, mis en place par le fournisseur.

On est donc bien là dans un cas où un chatbot peut « aller piocher » des informations dans des données, mais cela se fait par un système externe au LLM : ce que j'ai appelé le routeur, ainsi que le code qui gère la recherche internet. Ce système est opéré par le fournisseur du site web, et il résulte de capacités particulières (la production de tokens spéciaux pour dire "recherche web") et pour lesquels les LLMs ont été entrainés. Il me semblait important de préciser cette nuance !

Copyright © Blog Vénissian de Pierre-Alain Millet

Voilà, si demain votre cousin ou votre collègue affirme que "ChatGPT va chercher ce qu'il a lu sur des sites dans sa base de données", vous saurez quoi lui répondre !

Copyright © Blog Vénissian de Pierre-Alain Millet