Que veut dire « libre » (ou « open source ») pour un grand modèle de langage ?

Le flou entretenu entre open source et libre, déjà ancien et persistant dans l'industrie des technologies de l'information, revêt une nouvelle importance maintenant que les entreprises se lancent dans la course aux IA…
Explications, décantation et clarification par Stéphane Bortzmeyer, auquel nous ouvrons bien volontiers nos colonnes.

Vous le savez, les grands modèles de langage (ou LLM, pour « *Large Language Model* ») sont à la mode. Ces mécanismes, que le marketing met sous l'étiquette vague et sensationnaliste d'IA (Intelligence Artificielle), ont connu des progrès spectaculaires ces dernières années.
Une de leurs applications les plus connues est la génération de textes ou d'images. L'ouverture au public de ChatGPT, en novembre 2022, a popularisé cette application. Chaque grande entreprise de l'informatique sort désormais son propre modèle, son propre LLM.
Il faut donc se distinguer du concurrent et, pour cela, certains utilisent des arguments qui devraient plaire aux lecteurs et lectrices du Framablog, en affirmant que leur modèle est (en anglais dans le texte) « *open source* ». Est-ce vrai ou bien est-ce du « *libre-washing* » ?
Et qu'est-ce que cela veut dire pour cet objet un peu particulier qu'est un modèle de langage ?

Tout le monde parle des LLM (ici, avec une faute de frappe).

Source ouverte ?

Traitons d'abord un cas pénible mais fréquent : que veut dire « *open source* » ? Le terme désigne normalement l'information qui est librement disponible. C'est en ce sens que les diplomates, les chercheurs, les journalistes et les espions parlent de ROSO (Renseignement d'Origine en Sources Ouvertes) ou d'OSINT (*Open Source Intelligence*). Mais, dans le contexte du logiciel, le terme a acquis un autre sens quand un groupe de personnes, en 1998, a décidé d'essayer de remplacer le

terme de « logiciel libre », qui faisait peur aux décideurs, par celui d'« *open source* ». Ils ont produit une définition du terme qu'on peut considérer comme la définition officielle d'« *open source* ». Il est intéressant de noter qu'en pratique, cette définition est quasiment équivalente aux définitions classiques du logiciel libre et que des phrases comme « le logiciel X n'est pas libre mais est *open source* » n'ont donc pas de sens. Ceci dit, la plupart des gens qui utilisent le terme « *open source* » ne connaissent ni l'histoire, ni la politique, ni la définition « officielle » et ce terme, en réalité, est utilisé pour tout et n'importe quoi. On peut donc se dire « *open source* » sans risque d'être contredit. Je vais donc plutôt me pencher sur la question « ces modèles sont-ils libres ? ».

Grand modèle de langage ?

Le cas du logiciel est désormais bien connu et, sauf grande malhonnêteté intellectuelle, il est facile de dire si un logiciel est libre ou pas. Mais un modèle de langage ? C'est plus compliqué, Revenons un peu sur le fonctionnement d'un LLM (grand modèle de langage). On part d'une certaine quantité de données, par exemple des textes, le « *dataset* ». On applique divers traitements à ces données pour produire un premier modèle. Un modèle n'est ni un programme, ni un pur ensemble de données. C'est un objet intermédiaire, qui tient des deux. Après d'éventuels raffinements et ajouts, le modèle va être utilisé par un programme (le moteur) qui va le faire tourner et, par exemple, générer du texte. Le moteur en question peut être libre ou pas. Ainsi, la bibliothèque transformers est clairement libre (licence Apache), ainsi que les bibliothèques dont elle dépend (comme PyTorch). Mais c'est le modèle qu'elle va exécuter qui détermine la qualité du résultat. Et la question du caractère libre ou pas du modèle est bien plus délicate.

Notons au passage que, vu l'importante consommation de ressources matérielles qu'utilisent ces LLM, ils sont souvent exécutés sur une grosse machine distante (le mythique « *cloud* »). Lorsque vous jouez avec ChatGPT, le modèle (GPT 3 au début, GPT 4 désormais) n'est pas téléchargé chez vous. Vous avez donc le service ChatGPT, qui utilise le modèle GPT.

Mais qui produit ces modèles (on verra plus loin que c'est une tâche non triviale) ? Toutes les grandes entreprises du numérique ont le leur (OpenAI a le GPT qui propulse ChatGPT, Meta a Llama), mais il en existe bien d'autres (Bloom, Falcon, etc), sans compter ceux qui sont dérivés d'un modèle existant. Beaucoup de ces

modèles sont disponibles sur Hugging Face (« le GitHub de l'IA », si vous cherchez une « *catch phrase* ») et vous verrez donc bien des références à Hugging Face dans la suite de cet article. Prenons par exemple le modèle Falcon. Sa fiche sur Hugging Face nous donne ses caractéristiques techniques, le jeu de données sur lequel il a été entrainé (on verra que tous les modèles sont loin d'être aussi transparents sur leur création) et la licence utilisée (licence Apache, une licence libre). Hugging Face distribue également des jeux de données d'entrainement.

Dans cet exemple ci-dessous (trouvé dans la documentation de Hugging Face), on fait tourner le moteur transformers (plus exactement, transformers, plus diverses bibliothèques logicielles) sur le modèle xlnet-base-cased en lui posant la question « Es-tu du logiciel libre ? » :

```
% python run_generation.py --model_type=xlnet --
model_name_or_path=xlnet-base-cased
...
Model prompt >>> Are you free software?
This is a friendly reminder - the current text generation call
will exceed the model's predefined maximum length (-1).
Depending on the model, you may observe exceptions,
performance degradation, or nothing at all.
=== GENERATED SEQUENCE 1 ===
Are you free software? Are you a professional? Are you a
Master of Technical Knowledge? Are you a Professional?
```

Ce modèle, comme vous le voyez, est bien moins performant que celui qui est derrière le service ChatGPT ; je l'ai choisi parce qu'il peut tourner sur un ordinateur ordinaire.

Vous voulez voir du code source en langage Python ? Voici un exemple d'un programme qui fait à peu près la même chose :

```
from transformers import pipeline

generator = pipeline("text-generation", model="DunnBC22/xlnet-
base-cased-finetuned-WikiNeural-PoS")
print(generator("Are you free software?"))
```

Le modèle utilisé est un raffinement du précédent, DunnBC22/xlnet-base-cased-finetuned-WikiNeural-PoS. Il produit lui aussi du contenu de qualité

contestable([{‘generated_text’: « *Are you free software? What ever you may have played online over your days? Are you playing these games? Any these these hours where you aren’t wearing any heavy clothing?*) mais, bon, c’est un simple exemple, pas un usage intelligent de ces modèles.

Les LLM n’ont pas de corps (comme Scarlett Johansson dans le film « Her ») et ne sont donc pas faciles à illustrer. Plutôt qu’une de ces stupides illustrations de robot (les LLM n’ont pas de corps, bon sang !), je mets une image d’un chat certainement intelligent. Drew Coffman, CC BY 2.0, via Wikimedia Commons

Que veut dire « libre » pour un LLM ?

Les définitions classiques du logiciel libre ne s’appliquent pas telles quelles. Des entreprises (et les journalistes paresseux qui relaient leurs communiqués de presse sans vérifier) peuvent dire que leur modèle est « *open source* » simplement parce qu’on peut le télécharger et l’utiliser. C’est très loin de la

liberté. En effet, cette simple autorisation ne permet pas les libertés suivantes :

- Connaître le jeu de données utilisé pour l'entrainement, ce qui permettrait de connaitre les choix effectués par les auteurs du modèle (quels textes ils ont retenu, quels textes ils ont écarté) et savoir qui a écrit les textes en question (et n'était pas forcément d'accord pour cette utilisation).
- Connaître les innombrables choix techniques qui ont été faits pour transformer ces textes en un modèle. (Rappelez-vous : un algorithme, ce sont les décisions de quelqu'un d'autre.)

Sans ces informations, on ne peut pas refaire le modèle différemment (alors que la possibilité de modifier le programme est une des libertés essentielles pour qu'un logiciel soit qualifié de libre). Certes, on peut affiner le modèle (« *fine-tuning a pre-trained model* », diront les documentations) mais cela ne modifie pas le modèle lui-même, certains choix sont irréversibles (par exemple des choix de censure). Vous pouvez créer un nouveau modèle à partir du modèle initial (si la licence prétendument « *open source* » le permet) mais c'est tout.

Un exemple de *libre-washing*

Le 18 juillet 2023, l'entreprise Meta a annoncé la disponibilité de la version 2 de son modèle Llama, et le fait qu'il soit « *open source* ». Meta avait même convaincu un certain nombre de personnalités de signer un appel de soutien, une initiative rare dans le capitalisme. Imagine-t-on Microsoft faire signer un appel de soutien et de félicitations pour une nouvelle version de Windows ? En réalité, la licence est très restrictive, même le simple usage du modèle est limité. Par exemple, on ne peut pas utiliser Llama pour améliorer un autre modèle (concurrent). La démonstration la plus simple de la non-liberté est que, pour utiliser le modèle Llama sur Hugging Face, vous devez soumettre une candidature, que Meta accepte ou pas (« *Cannot access gated repo for url https://huggingface.co/meta-llama/Llama-2-7b/resolve/main/config.json. Access to model meta-llama/Llama-2-7b is restricted and you are not in the authorized list. Visit https://huggingface.co/meta-llama/Llama-2-7b to ask for access.* »)

Mais la communication dans l'industrie du numérique est telle que très peu de gens ont vérifié. Beaucoup de commentateurs et de gourous ont simplement relayé la propagande de Meta. Les auteurs de la définition originale d'« *open*

source » ont expliqué clairement que Llama n'avait rien d'« open source », même en étant très laxiste sur l'utilisation du terme. Ceci dit, il y a une certaine ironie derrière le fait que les mêmes personnes, celles de cette Open Source Initiative, critiquent Meta alors même qu'elles avaient inventé le terme « open source » pour brouiller les pistes et relativiser l'importance de la liberté.

Au contraire, un modèle comme Falcon coche toutes les cases et peut très probablement être qualifié de libre.

La taille compte

Si une organisation qui crée un LLM publie le jeu de données utilisé, tous les réglages utilisés pendant l'entrainement, et permet ensuite son utilisation, sa modification et sa redistribution, est-ce que le modèle peut être qualifié de libre ? Oui, certainement, mais on peut ajouter une restriction, le problème pratique. En effet, un modèle significatif (disons, permettant des résultats qui ne sont pas ridicules par rapport à ceux de ChatGPT) nécessite une quantité colossale de données et des machines énormes pour l'entrainement. L'exécution du modèle par le moteur peut être plus économe. Encore qu'elle soit hors de portée, par exemple, de l'ordiphone classique. Si une application « utilisant l'IA » tourne soi-disant sur votre ordiphone, c'est simplement parce que le gros du travail est fait par un ordinateur distant, à qui l'application envoie vos données (ce qui pose divers problèmes liés à la vie privée, mais c'est une autre histoire). Même si l'ordiphone avait les capacités nécessaires, faire tourner un modèle non trivial épuiserait vite sa batterie. Certains fabricants promettent des LLM tournant sur l'ordiphone lui-même (« o*n-device* ») mais c'est loin d'être réalisé.

Mais l'entraînement d'un modèle non trivial est bien pire. Non seulement il faut télécharger des téra-octets sur son disque dur, et les stocker, mais il faut des dizaines d'ordinateurs rapides équipés de GPU (puces graphiques) pour créer le modèle. Le modèle Llama aurait nécessité des milliers de machines et Bloom une bonne partie d'un super-calculateur. Cette histoire de taille ne remet pas en question le caractère libre du modèle, mais cela limite quand même cette liberté en pratique. Un peu comme si on vous disait « vous êtes libre de passer votre week-end sur la Lune, d'ailleurs voici les plans de la fusée ». Le monde du logiciel libre n'a pas encore beaucoup réfléchi à ce genre de problèmes. (Qui ne touche pas que l'IA : ainsi, un logiciel très complexe, comme un navigateur Web, peut

être libre, sans que pour autant les modifications soit une entreprise raisonnable.) En pratique, pour l'instant, il y a donc peu de gens qui ré-entrainent le modèle, faisant au contraire une confiance aveugle à ce qu'ils ont téléchargé (voire utilisé à distance).

Conclusion

Pour l'instant, la question de savoir ce que signifie la liberté pour un modèle de langage reste donc ouverte. L'Open Source Initiative a lancé un projet pour arriver à une définition. Je ne connais pas d'effort analogue du côté de la FSF mais plus tard, peut-être ?