LIDILE Linguistique Ingénierie et Didactique des Langues

"Understanding the 'Large Language Models' " - Dr. Cyriel Mallart

Understanding the « Large Language Models » : Where do they come from and where can we go with them?

In this presentation, we will cover the fundamental concepts of machine learning without requiring an in-depth knowledge of mathematics. We will start by defining, at a high level, what a model is and explain the steps involved in training a machine learning model, highlighting the key principles and associated challenges. We will then focus on large language models (LLMs), explaining why these models are particularly powerful and what distinguishes them from other approaches to automatic language processing. We will discuss their capabilities in terms of language representation, to explain their potential for text generation, contextual understanding and, above all, adaptation to a variety of tasks. Finally, we will examine the potential applications of LLMs, highlighting the different families of LLMs and the tasks traditionally associated with them. The main aim of this presentation is to equip experts in linguistics with the vocabulary and concepts essential for effective dialogue with experts in machine learning, in order to accelerate research in both fields through the cross-fertilisation of ideas and perspectives

Monday 27 May 2024 - 2PM - Room L201

Cyriel Mallart
Research engineer in automatic language processing - LIDILE

To join online:
[http://bit.ly/lundi_lidile | http://bit.ly/lundi_lidile]
Meeting ID : [callto:873 5180 7764 | 873 5180 7764]
Secret code : Evain

Supported by A4U

Textual transcrition of the conference in French

Bonjour à tous et à toutes, merci d'être présents ici en présentiel et en ligne. Aujourd'hui, on a le plaisir d'écouter Cyriel Mallart, qui est actuellement ingénieure en recherche sur le projet A4LN que je te l'ordonne. Cyriel a un background historique de formation, de statisticien, d'informaticien, de data science. Elle a fait sa thèse à l'INRIA sous la direction de Pascal Cébillot et de Guillaume Gravier. Gros suivi précis et rigoureux. C'était à donner une thèse sur la classification de données textuelles dans le journal de West France. Et dans le cadre de ce travail-là, Cyriel a une expertise avérée dans différents types de systèmes par apprentissage supervisés, donc de l'intelligence artificielle ancienne génération, on va dire, et une grosse expérience aussi sur les systèmes d'intelligence artificielle qui apparaissent, les large language models, avec le premier système qui est apparu en 2018 chez Google, le BERT. Donc on s'est dit que ce serait très intéressant pour nous, les linguistes, de comprendre un peu ce qu'il y a sous le capot. Ce n'est pas simplement de regarder ce qu'ils font, mais c'est aussi de se préoccuper de questions de comment ils le font, sur une certaine mesure, comment ces décisions sont prises, comment on peut les adapter, comment est-ce qu'on les réutilise, de manière à ce que nous, linguistes, puissions nous poser des questions plus approfondies que le simple constat de ce que le texte système nous produit. Et je pense que ce que Cyriel va nous montrer, c'est qu'il y a plein de façons de les utiliser, de les adapter, de les réentraîner, dans lesquelles des linguistes peuvent jouer un rôle. Non pas que les linguistes peuvent tout faire, puissent tout faire, ce n'est pas l'idée, mais c'est de comprendre que le linguiste peut s'insérer dans une équipe de thalistes, d'informaticiens, de statisticiens, pour produire des systèmes qui s'inspirent, entre autres, de problématiques linguistiques, et pas seulement, pour avoir des choses beaucoup plus complexes, mais qui s'adaptent du point de vue linguistique. Donc, Cyriel, merci d'avance. On a hâte. Allez, c'est parti. Je vous avais promis, il y a Titine qui est ici. Donc, bonjour. Alors, Thomas nous a promis beaucoup de choses, une journée, on va faire un petit peu moins que prévu, mais dans l'idée, on reste sur la même ordre d'idée, qui est un peu de comprendre les large language borders, et surtout d'où il s'agit. Donc, j'ai un petit coup. Voilà. Alors, pour ce qu'on crée l'antiquaire, parce que je ne savais pas exactement quel public on allait avoir, j'ai un petit questionnaire présent, à savoir ce que nous attendait cette présentation. Déjà, pour que je puisse actuellement adapter la présentation. Alors, vraiment, c'est la question. Moi, je n'ai pas d'autre question, mais si vous ne voulez pas, vous pouvez me le mettre. Alors, c'est fini. Pour découvrir simplement ce que c'est qu'il m'a dit dernièrement, si vous n'avez pas vraiment pas l'idée de ce que ça doit être, si vous pensez plus sur la structure de l'éthique de l'Anse, ou sur l'encherie du Cotidium, vu que ça sort, peut-être qu'il y a des questions un petit peu plus pratiques. Je ne sais pas. Donc, on va rester sur un truc un petit peu plus léger par rapport au concept de machine learning. On va surtout parler des LLM et de leurs applications, je pense, pour aujourd'hui. Surtout si vous avez déjà des questions. Très bien. Voilà, ils savent, mais évidemment. Évidemment, il va falloir quand même que je vous parle de quelque chose qui fait un peu mal. Donc, pour savoir un peu ce dont on va parler aujourd'hui, vous avez deux ou trois objectifs, savoir où on va, en fait, à la fin. J'avais quand même un premier objectif, c'est de prendre des bases de machine learning, parce que sans comprendre ça, on n'arrive pas à comprendre leurs règles. Ensuite, les enjeux de l'entraînement, pas savoir comment ça s'entraîne, savoir ce que sont les grands défis que ça peut donner. Les LLM, c'est une fameuse particulière des modules de machine learning, de voir à quoi c'est différent et surtout, si on veut pouvoir les utiliser et où on commence. Désolé pour ceux qui sont allergiques, en premier on a des maths. Pour nous, ce n'est pas les maths du format. Moi, je suis allergique au math aussi, donc on va laisser ça assez léger. Ensuite, un crash course, mais encore une fois, on pense que le mieux c'est de trouver ça sur les machines learning. Et ensuite, on va surtout parler des LLM et donner un peu de temps, en cas de défense, créer des préconisations, les possibilités que ça peut donner, et les sous-titres là aussi. Donc, c'est plus rapide, les maths, vous avez le droit de déconnecter le cerveau pendant dix minutes, il n'y a pas de soucis, ça marche aussi. Parce qu'en fait, les machines learning, je suis désolée, c'est que les maths. Des maths, mais avec des ordinateurs. Donc, pour rafraîchir les cerveaux de ceux qui sont vraiment à temps, on va parler en fait plus d'un peu plus à haut niveau que Nathanael. En gros, vous imaginez ce que c'est qu'une fonction. Si vous avez une usine, vous prenez des pièces, vous les passez dans l'usine et ça vous sort un produit. Ici, je prends l'exemple d'une voiture, vous notez un volant, un moteur, des roues, ça vous fait une voiture à l'air. Et bien, c'est exactement le même principe en mathématiques que le principe de la fonction. Vous avez une entrée, vous lui faites des choses, vous n'êtes même pas forcément obligé de savoir ce qu'il y a à l'intérieur, mais vous lui faites des choses, ici par exemple une multiplication, et vous en trouvez un résultat. Vous gardez à l'esprit une entrée, un traitement et un résultat. Et finalement, quand on commence à faire avec des lettres, ça devient exactement la même chose. Une entrée qu'on appelle souvent X, une fonction qu'on appelle souvent F et une sortie qu'on appelle souvent X. Jusque-là, on est bon. On a une usine qui fait des traits. Le problème, c'est quand on veut commencer à comprendre ce qui se passe dessous. C'est-à-dire, j'ai mon usine, mais je ne sais pas quelles sont mes règles. Si je mets trois roues et un volant, qu'est-ce qui se passe ? Si je mets un moteur V8, ça me sorti le moteur K. Donc, ce n'est pas exactement dans quel sens ça. Et c'est pour ça qu'on parle d'utilisation. Peut-être que vous avez une liste de règles, c'est un peu plus compliqué, mais quand vous avez une entrée, vous avez une idée de la sortie que ça va avoir, en général parce qu'on connaît la fonction. On connaît en fait le manuel d'utilisation de l'usine. Et par exemple, si on rentre entre l'utilité de trois roues, un moteur, et par beaucoup de puissance, on a une Ryan Chobin dans la voiture du service. Sinon, si on commence à mettre beaucoup plus d'engins, beaucoup plus de puissance, beaucoup plus d'argent, on se retrouve avec quelque chose d'un peu mieux. Et en fait, tout le problème qu'on va avoir, notamment dans le machine learning, c'est de trouver à l'entrée quelle est la sortie. Et notamment, je parle de l'optimisation. L'optimisation, c'est un mot qui va beaucoup ressortir, donc je suis désolée que je le dise tout le moment. Simplement, quand on a une fonction, quand on sait comment une usine fonctionne, comment on fait pour maximiser son emploi ? Comment on fait en ayant un certain nombre de pièces, un certain nombre, une certaine quantité d'argent ? Comment est-ce qu'on fait pour produire la voiture la plus qualitative, on va dire, en ayant tout ça ? Et c'est ce qu'on appelle de l'optimisation. Et c'est en fait le principe de base du machine

learning. On a des fonctions, et on les optimise. Allez, il y a de nombreuses étapes. On va partir du problème du machine learning avec un peu moins de temps. D'abord, c'est là où j'allais faire un deuxième petit. Si vous avez encore le bouton du beurre, c'est là où j'allais faire une deuxième question. Vous allez me dire, quand je vous dis machine learning, qu'est-ce que vous pensez ? Vous pouvez me dire en tête quelles sont les préoccupations que vous avez, même s'il y a quelque chose qui a loupé. J'imagine que c'est l'automatisation. Ok, on a l'émotion des objets, on a l'impression. Et bien, on a l'impression. Et bien, c'est vraiment pas mal, parce que les lignes peuvent être en sorte. Et bien, on a l'impression. Et bien, derrière, on a l'impression de la liste. Bon, je vais arrêter là, parce que moi, j'ai l'impression que ce n'est pas très très bien. C'est super. Apprentissage, très bien. Et bien, vous m'avez parlé de quasiment tout ce dont je vais vous parler dans les prochaines minutes. C'est bien, on va pouvoir remettre les notions en place ensemble. Ou alors, si je ne vais pas assez vite, n'hésitez pas à me le dire. Alors, ce que je m'attendais à entendre, et que du coup vous n'avez pas vu, et c'est bien parce que ça me permet de raconter une notion, c'est ce que c'est qu'un modèle. Parce qu'en fait, derrière le machine learning, il y a toujours un modèle. En fait, on dit un modèle de machine learning. Et un modèle, c'est quoi en fait ? C'est une description d'un phénomène. Souvenez-vous de notre usine, quand on montre un truc, il y a une transformation qui se fait, il y a un truc qui sort. Et en fait, la transformation, c'est ce modèle. Ça ne vient pas du nul part. Donc on part du principe qu'on connaît l'univers. On décrit un certain phénomène avec certaines règles. On dit, en règle générale, ça marche comme ça. Et on le décrit de façon mathématique. Juste dire, par exemple, fois trois, c'est aussi un modèle mathématique. Dire, je prends trois bières, ça coûte donc six euros fois trois, c'est un modèle mathématique. On a une entrée, un traitement, une sortie. C'est juste un modèle très simple. Donc voilà. Donc je comprends un peu ce que je viens de dire. Les entrées, ce qu'on nous sert, les pièces qu'on met. Les sorties, c'est comment le phénomène évolue, c'est-à-dire avec des pièces, on fait une voiture. Et maintenant, il y a deux notions auxquelles il va falloir faire l'invention. C'est la notion du modèle, c'est-à-dire l'usine. En gros, qu'est-ce qu'on fait ? Mais aussi une notion plus fine, qui est la notion de paramètres. La notion de paramètres, en fait, c'est le manuel. Quand vous avez une usine qui fait des voitures, vous en faites plein de voitures, plein de différentes. Vous avez de lave de chevaux. Je ne sais pas, mais vous êtes chez Citroën, ça vous fait tout, de la Clio à... Je ne sais pas pourquoi je fais pas Citroën, mais ça vous fait beaucoup de choses. Et en fait, les paramètres, c'est, avec ce qu'on rentre, quelles sont les règles, en fait, quel est le manuel pour faire pire voiture à la fin. C'est un peu ça, les paramètres. Donc, en gros, vous avez un modèle, c'est une description de comment les choses marchent. En général, un véhicule, c'est un machin avec un moteur, un carrossier, quatre roues. Les paramètres, c'est comment on va les mettre dedans. Dans quel sens ? Et donc, en fait, dans un modèle, le modèle est fixé. C'est le paramètre qu'on va essayer. Donc, déjà, à quoi ça sert d'avoir un modèle ? Et aussi, ça sert de voir un des paramètres. Un modèle, vous les voyez tous les jours, en fait, et vous ne s'y faites pas attention. La météo, par exemple, c'est un modèle, sachant les percures d'argent, la période de l'année, l'endroit où on est. On a une idée, en fait, en général, de comment ça va se passer, de ce qui va arriver. Dans le monde financier pareil, plus ou moins une certaine quantité d'incertitude, ou même quand vous avez des algorithmes de recommandations sur votre service de streaming préféré, c'est aussi un modèle qui, connaissant qui vous êtes, quel type de consommateur vous êtes, va vous prédire des films à suivre. Un truc important à comprendre, c'est que le modèle, il ne sort pas de nulle part. Et surtout, il y a une hypothèse qui est faite derrière, c'est que le monde continue à respecter les mêmes règles. Une idée, c'est, par exemple, avec l'idée de la météo, on commence à avoir des modèles qui ne sont plus fiables parce qu'il y a eu trop d'évolutions par rapport au gold stream et au courant. En fait, un modèle n'a de valeur que si on est encore dans le même type de situation qu'on l'était avant. Quand on construit un, il faut faire attention à la situation dans laquelle on est, et donc à son adaptation, ou pas, à d'autres situations. Alors, je vous parle de modèle, je vous parle qu'il y en a partout, mais d'où ça sort ? Il y a deux étapes quand on essaie de modéliser un phénomène. La première, c'est, ah, j'ai un phénomène, je dois modéliser. Je veux trouver quelle est la règle en général qui régit ce phénomène. La première étape, c'est de trouver le modèle, c'est-à-dire de trouver quelle est en général la forme que va suivre ce phénomène. Est-ce que je veux dire où ça fait une voiture, ou est-ce que je suis en train de produire des bateaux ? Je ne sais pas. Il faut trouver la forme générale. Pour ça, on a deux choix. Soit on est des experts et on sait qu'on a des modèles qui marchent très bien. On fait des séries temporelles quand on a, par exemple, du trafic dans les aéroports, ce genre de choses. On le sait, et c'est un peu l'argument d'autorité. En général, on fait comme ça, et très souvent, ça marche bien. Une autre solution, c'est d'en essayer plein et de voir celui qui marche mieux. Disons qu'on a trouvé notre modèle. Quelqu'un vous a dit, là, je te jure, c'est ça, fais-moi un peu plus. Le problème maintenant, c'est de trouver les bons paramètres, c'est-à-dire comment ce modèle va faire pour modéliser proprement, c'est-à-dire comment il s'adapte à la situation dans laquelle on est. Et c'est tout ça le problème du machine à mot, c'est de trouver les bons paramètres. Parce qu'ils ne sortent pas de nulle part, on ne peut pas les inventer. Alors oui, quand il s'agit d'Y, c'est bien de trouver quelque chose, on peut les calculer à la main. Mais quand on commence à avoir beaucoup de données, des données météorologiques, des données, tout ça, la tendance est absolument gauche. Le grand enjeu d'un modèle de machine à mot, ça va être de découvrir les règles, autrement dit, d'estimer les paramètres. Et pour estimer les paramètres, il n'y a pas mieux que de se dire qu'est-ce qui se passe dans la vraie vie ? Je compare mon modèle à ce qui se passe et j'apprends comme ça. Et donc, arrive la notion principale qui est la notion de données. Il faut utiliser les données par exemple. Si vous voulez reprendre un peu la métaphore de tout à l'heure, c'est vous êtes un alien qui arrive sur Terre, qui voit une usine, qui voit qu'il y a un truc qui rentre et un truc qui sort de l'usine. Il y a un peu une idée générale que c'est une usine à faire des véhicules, mais du coup maintenant il va falloir comprendre quelles sont les règles en regardant les choses rentrées, les choses sorties, et sur son petit carnet dire, ça, ça fait ça, ça, ça fait ça. Maintenant, c'est à nous de les faire. Et ce qui mène les paramètres, ça se passe. Il y a une zone de cycles qui se passe pour entraîner des modèles de machine à mot comme ça. En fait, on a une situation réelle, on a notre modèle, et on applique ce modèle à la situation réelle, on regarde ce qui sort. Je veux dire, il y a quatre roues, un chat, je fais une voiture. Quelle voiture ça sort ? Et en fait, je vérifie par rapport à ce que l'on sait, est-ce que c'est vraiment la bonne chose ? Est-ce que j'ai bien deviné ce qu'il allait en sortir ? Si j'ai pas bien deviné, et bien simplement je vais changer un peu mes règles, je vais modifier quelque chose, et puis je recommence. Donc comme ça, on a l'impression qu'il y a un peu de noir. C'est la forme des règles qui permet à la climatique, les modèles des modèles, et puis quand même d'aller dans le jeune film, je peux changer un petit peu mes règles à chaque fois. Alors là, on parlait d'un élan qui m'a beaucoup parlé au début, quand j'ai vu qu'on pensait que c'était qu'un entraînement de résumé norme. On va absolument faire complètement toutes les notions mathématiques en scie. On a un modèle, vous avez même un petit rouge, il s'appelle modèle, il s'appelle notion. Et ce qu'on veut, c'est qu'un modèle, à la fin de sa vie, arrive à distinguer entre des photos de chats et des photos de Muffy. Et là, on peut jouer. Je vais vous présenter deux autres robots. Le robot créateur. Le robot créateur, le but de sa vie, c'est de créer des petits robots, de créer des modèles. Le truc, c'est qu'il doit mettre les composants et les lier pour faire passer l'électricité, etc. Et ses composants, il les lie. Au début, il ne sait absolument pas ce qu'il fait. Il a des composants, il a des fils. Il regarde, il fait bombe. Il ne le kiffe, parce qu'il ne sait pas du tout ce qu'il fait. Alors il essaye encore d'autres. On est d'accord, il ne va pas en faire qu'un seul. Il va essayer plein de versions de petits modèles comme ça, où il va brancher différemment et puis regarder ce qu'il fait. Il branche. Il est content. Une fois qu'il a fait ses branchements, il y a un autre robot qui arrive. Le robot propre. Alors, pas vraiment un prof, c'est plus un preneur d'examen, mais même pour la simplicité, le robot propre, qui, quand il a un petit modèle en face de lui, il lui demande, est-ce que c'est un chat ou est-ce que c'est un Muffin ? Et il prend la réponse. Ensuite, tout ça va obéir à un site. Le robot créateur, il va faire son petit robot, un peu vide. Il va l'envoyer à l'école. Au test, plus exactement. Là, le robot propre, il demande, est-ce que c'est un chat ou est-ce que c'est un Muffin ? Là, il y a deux choix. Soit il réussit. Au cas où, c'est un peu cruel. On le revoit au robot créateur qui lui ouvre le crâne et qui va regarder comment il est fait. C'est-à-dire, celui-là a marché, comment j'ai fait pour que ça marche ? Après, il filme dans la pile. Dans le deuxième cas, le petit robot, le petit modèle n'est pas réussi. Il finit à la poubelle. Et ça, le robot créateur, il ne va pas en faire qu'un des petits modèles. En gros, il va faire toute une classe. Mais au début, n'importe comment. Mais au fur et à mesure où il rouvre le cerveau de tout le monde et il voit comment c'est fait, il reproduit ce qui s'est passé. C'est-à-dire, celui-là, il a bien réussi. Hop, je vais recommencer ce branchement parce que je l'avais l'air d'avoir à peu près bien réussi. Au début, les branchements sont tous boiteux, mais de moins en moins, en fait. Et donc, à chaque fois qu'il envoie toute une classe à l'école et qu'il y en a un certain nombre qui est aussi mieux que les autres, eh bien, on prend ce curieux RSI et on roule tout le reste. Et en fait, ce cycle se répète. Donc, à chaque fois, on sélectionne les meilleurs, on regarde ce qui se passe dans leur tête et on fait les prochains comme on avait fait les premiers. Tout ceci, ça fait sur des milliers de générations. Il n'y en a pas qu'une, parce que sinon, on ne va pas assez vite. Et donc, toutes les générations, on les envoie à l'école, on regarde comment ça se passe et on reprend les meilleurs. Par contre, il peut se passer une chose, enfin, il va se passer une chose au bout d'un moment, c'est que tout le monde va être très bon. Et bien, qu'est-ce qu'on fait dans ce cas-là ? Eh bien, on prend les meilleurs et on envoie les autres au rubis. Au bout d'un moment, en fait, tout commence à être très bon et on prend les meilleurs de la classe et ça commence à être impolé. Mais il faut se foutre un peu ceux qui ont trop essayé, trop dur. Et c'est à ce moment-là qu'on s'arrête parce que tout est devenu très bon et qu'on n'a plus besoin de griller, etc. Et c'est là où le robot créateur, lui, va enfin comprendre et bien ça y est, pour réussir cette tâche-là, pour réussir à créer un petit modèle qui distingue bien entre un chien et un muffin, il faut exactement ce branchement-là. Et c'est bon. Alors, je vous ai fini le modèle en très longtemps, mais en fait, tout ça, ça vit vraiment dans la vie de tous les jours. Vous avez peut-être rencontré nos petits robots-là. Donc, toutes nos versions d'un truc, à chaque fois qu'on leur passe un test, c'est un batch en fait. Ça, si vous avez entendu la notion, c'est ça. En gros, on leur donne un exemple, ils y vont, ils se plantent ou pas, mais ils donnent une information. Le prof, c'est une fonction de perte, c'est un élément très important. C'est ce qui nous permet de dire qu'est-ce qui est beau et qu'est-ce qui n'est pas beau, qu'est-ce qu'on attendait à la fin et qu'est-ce que le modèle a fait et est-ce que c'est la même chose. C'est une fonction très importante. Le robot créateur, c'est ce qu'on appelle un optimisateur, un optimizer en anglais. Vous vous souvenez de notre fonction là ? Eh bien, c'est lui en fait, parce qu'il va regarder qui a été bon. Il va regarder les meilleurs et reproduire. Et le phénomène de renvoyer les robots pour se faire ouvrir quand ils sont bons, c'est ce qu'on appelle la basse propagation. Donc, si vous avez besoin de retrouver ces notions-là, c'est un peu ce principe. Alors, quelques mots de précaution. Parce qu'il y a un phénomène qui se passe,

c'est que pourquoi un câblage marche mieux qu'un autre ? On ne sait pas. En fait, on ne sait pas pourquoi ça marche mieux. On sait juste que ça marche mieux. Le robot créateur, il ouvre les cerveaux, il va regarder, il revanche, mais en fait, il n'a aucune connaissance de ce qui devrait être là ou de ce qui marche mieux. Il prend juste ce qui marche et il recommence. Donc en fait, en machine learning, et c'est vrai dans tous les cas, on ne sait pas pourquoi un certain modèle, entraîner cette personne-là, ne marche mieux qu'un autre. En général, il y a des choses à expliquer à l'heure actuelle, mais pourquoi ça a pris ces poids-là ? Il n'y a pas de vérité théorique qui vous dit qu'il fallait que ça soit le droit et le clair. Il y a aussi une deuxième notion très importante, c'est que le robot prof, c'est lui qui décide qui meurt ou qui vit. Et donc, le test qu'on lui donne, c'est-à-dire, là, c'est un chat, là, c'est un racine, en fait, c'est la chose la plus importante qu'on puisse avoir, puisque c'est lui qui décide, enfin, en fait, qui on garde. Et donc, les tests créés, c'est-à-dire vos données d'entraînement, c'est le truc le plus important que vous pouvez mettre dedans. Parce que oui, petit modèle, il apprend des choses, c'est branchement différent, mais si vous envoyez au rebut des modèles qui seraient bons, en fait, vous n'arrivez pas à obtenir quelque chose de bon à la fin. Et enfin, on l'a vu dans le dernier exemple, s'arrêter, ça devient un art, parce qu'on n'a pas d'autre règle pour dire on s'arrête. On en a maintenant, mais on n'a pas de règle mathématique absolue qui vous dit, à ce moment-là, ça va, on arrête. On a des idées, c'est-à-dire, si 98 % des élèves passent, on s'arrête, ou si on a fait 15 générations, on s'arrête. Mais en fait, on n'a pas de règle pour dire, si celui-là, celui-là, celui-là, il passe, on s'arrête, et on s'arrêtait, ça devient un mode, puisqu'au bout d'un moment, on obtient des robots qui sont très, très bons, mais trop bons, à seulement une certaine tâche, une certaine chose. Et donc, il faut savoir s'arrêter avant. Donc, tout ça, ça nous amène aux raisons de neurones, parce que je n'ai pas dit ça pour rien. En gros, il y a des raisons de neurones, mais il y a des découpes au milieu, c'est plus qu'à une tâche, c'est exactement la même chose que le cerveau de notre petit modèle. Alors, on a des neurones, et vous pouvez assimiler un neurone à une fonction mathématique, il y a une entrée, il y a une sortie, il y a un truc qui se passe au milieu. Pour ceux qui sont mathématiquement inclinés, c'est une fonction linéaire suivie d'une fonction non linéaire, mais en général, c'est juste des masses. C'est une grosse fonction mathématique. Et en fait, on les passe des unes aux autres. La raison pour laquelle on a tasse quand même, qu'on n'ait pas à s'arrêter, c'est parce qu'en fait, c'est beaucoup plus simple de gérer les choses de façon, de faire des économies d'échelle en informatique. C'est-à-dire, au lieu d'avoir la réponse à celui-là, la réponse à celui-là, la réponse à celui-là, on agrège tout le monde, croque en carré, les grosses matrices, et ça optimise nous. D'autant plus que plus c'est gros, plus on peut avoir de détails. Et le détail, c'est ce qui nous intéresse, parce que ça permet d'avoir des préditifs très finis. Par exemple, sur le GPT, on est sur le milliard de paramètres, c'est-à-dire des millions de milliards, des millions comme ça. Si vous voulez voir à quoi ça ressemble, ce n'est pas exactement dans la vraie vie. Ici, c'est un espace de jeu TensorFlow. C'est des librairies, il y a beaucoup d'informatiques qui sont disponibles, notamment, et en gros, vous voyez exactement le même truc. Vous avez un problème. Ici, la question, c'est de dire qui sont en haut et qui sont bleus. Là, vous avez votre modèle avec ses câblages en interne, ses entrées, tous ces nouveaux qui le donnent de sorte. Et on va rejouer l'entraînement. En fait, vous voyez qu'au fur et à mesure, tout doucement, ça converge vers une solution qui vous dit qu'en général, au-dessus de ça, c'est orange et en derrière, c'est bleu. Et en fait, ça se traduit dans les neurones par une certaine forme. Donc, en fait, les neurones, c'est-à-dire, ils ont acquis une connaissance à ce petit élément pour aider à classifier les choses. Vous voyez ici, on voit que c'est tout droit ou que c'est à l'autre des biais. Mais en fait, c'est l'ensemble de ces éléments, c'est l'ensemble de ces petites définitions individuelles qui vont faire qu'on va pouvoir prendre une décision à l'heure. Donc, si vous voulez remettre ça en place, on a des entrées, on a des sorties et au milieu, ça apprend des proies, des trucs mathématiques, ça apprend des choses qui permettent de câbler bien pour avoir une sortie. Et en plupart du temps, on pense que ce qu'il y a à l'intérieur, ici, c'est une représentation cachée de la vérité, de ce qui se passe dans la vie, du phénomène. Et ça, on va laisser une représentation cachée, notamment parce que quand on n'y a pas accès très facilement. Donc, ici, une représentation cachée parce qu'en fait, c'est un tas de maths, c'est un tas de chiffres qui, finalement, vous donnent des indications qui résument un peu le phénomène que vous avez observé. Ici, par exemple, si on prend ça, à la fin, on a une dernière fonction, une liste d'agrégation, et à la fin, ça vous dit oui ou non, c'est l'achat. C'est un chat ou c'est un muffin. Mais en fait, si on a cette représentation cachée, ça, en fait, on en a plus de bonheur, on peut la retrouver facilement. On dit oui, c'est un chat ou un muffin. En plus, on a appris quelque chose d'encore plus fond que l'oubli, mais non, on a appris quels sont les endroits, les moments dans une image qui font que c'est un chat ou un muffin. Donc, c'est ce qu'on appelle du transfert learning. Il est possible d'enlever la dernière couche, la couche qui vous dit oui ou non, de garder cette représentation au milieu et de rajouter une nouvelle couche qui vous intéresse. Par exemple, maintenant, est-ce que c'est un avion, un bateau, un chat ou un muffin ? Donc, on peut en fait, c'est très restreint comme tâche, mais on peut en fait changer la tâche finale. Et le truc bien, c'est que tout ça, là, ça reste entraîné. C'est-à-dire, on a quand même une idée des proies. On a quand même une idée d'avoir vu avant le phénomène de chat ou de muffin. Et donc, on a déjà cette connaissance et cette information. Donc, on sait que le modèle, il est déjà plutôt bon à reconnaître ça. Et donc, on peut faire d'autres choses en partant cette connaissance précédente. Et notamment, on va avoir besoin de moins entraîner, parce qu'on sait déjà que le modèle, quand il va à l'école, il n'est déjà pas trop bon d'être. Donc, il reste qu'à le ré-entraîner sur des choses qu'il connaît un peu moins. Alors, est-ce que... Parce que depuis tout à l'heure, je vous parle de muffin et des chats. Est-ce que c'est toujours ce qu'on fait ? Il y a plusieurs objectifs, c'est-à-dire plusieurs tâches, c'est-à-dire plusieurs tests que peut donner le robot prof. Classification, qu'est-ce qu'on fait, qu'est-ce que c'est, par exemple dans le niveau CEFR A1, B1, A1, A2, B1, B2, mais on peut aussi faire d'autres choses, la régression, en donnant une valeur, une valeur qui ressort, par exemple comme on a vu dans le code, si je donne le nombre de muffins que je vais acheter, ça me donne le prix que je vais payer et on peut apprendre en fait la fonction de prix. On peut aussi générer des choses, des images, du texte, en donnant quelque chose, on donne n'importe quoi, du bruit, ça nous génère quelque chose, et enfin on peut gérer des groupes, c'est-à-dire au lieu de dire ça c'est un chat, ça c'est un muffin, on lui apprend à faire des groupes en lui donnant des images en disant ben en plus ça c'est un groupe, ça c'est un groupe de muffins, et il apprend à faire des groupes. Des entraînements, la plupart du temps on va être honnête, ils sont supervisés, c'est-à-dire on a une vérité, ça c'est un chat, ça c'est un muffin, mais il y a d'autres possibilités, notamment avec le texte, on fait du non-supervisé, c'est-à-dire qu'on n'a pas besoin de quelqu'un qui s'assoit et qui met oui ou non sur une sur une étiquette, et ça c'est très puissant, parce que c'est moins coûteux, beaucoup moins coûteux, il n'y a pas de chercheur qui veut s'asseoir à son bureau tous les soirs à faire 40 trucs. Après, les techniques sont pas forcément toutes aussi bien, mais les résultats parfois sont différents. Et du semi-supervisé qui est un peu une combinaison des deux, on a quelques exemples à noter, on en déduit les exemples à noter, et la dernière chose qui n'est pas vraiment très intéressante pour nous, mais parfois forcément c'est un peu comme les robots qui apprennent à marcher, vous voyez, il fait un premier pas, il se casse la figure, il apprend que ce n'est pas comme ça qu'il devait faire, un peu comme un enfant apprend, donc c'est un peu plus compétit comme type d'entraînement, et je ne vais pas rentrer dans l'ordre. Mais retenez qu'il est possible de faire plus l'une de la classification, et que les bruitises d'apprentissage principaux, de supervisé et de non-supervisé. Donc, on a un entrainement, c'est un processus spératif, qu'il y a besoin de donner. le modèle fait ses choix, on ne sait pas, on sait juste qu'il est fait bien ou pas. Et enfin, on peut changer la tâche finale à condition d'avoir de quoi réentraîner sur la nouvelle tâche. Mais ce sera moins cher. Donc, si vous en avez marre, si vous avez décroché un moment, il n'y a que ça à souligner, que dans un réseau on a une représentation cachée, on a une grosse matrice qui représente ce qui est rentré dedans, que ça passe par un réseau neurone, qu'on a la capacité de générer des choses avec un réseau neurone, c'est-à-dire d'avoir une sortie qui ne soit pas du tout liée à ce qu'on a rangé, qu'il est possible d'enlever un morceau à la fin et d'en rajouter une nouvelle sortie et enfin qu'il est possible de le faire sans avoir besoin de donner à noter. Ça va faire du son dans 15 secondes, parce que tout ça, c'est ce dont on a besoin pour parler des large language models. Alors, comme tout le monde l'a dit, au début, c'est la version papa-maman du machine learning, c'était l'entrainement supervisé, ça c'est la version 2020, mais attention, ça ne reste qu'un cas particulier. Ce n'est pas la panacée, ça marche que dans certains cas, et très bien dans ces cas-là, mais ce n'est pas les seules choses qui existent. Donc, un MNM, c'est un réseau neurone, mais particulier. Je ne vais pas passer dans le sens, déjà le premier truc, c'est que ça ne fait que du langage. Il n'y a pas d'image, il n'y a pas de texte, il n'y a pas de son, donc en fait il faut réussir à transformer tout ce qu'on met dedans en texte. Le deuxième, c'est que c'est un réseau de langage, mais en fait, il cherche une représentation des règles de la langue. Il ne cherche pas forcément à générer une classification où quand je vois ça, forcément c'est positif, quand je vois ça, c'est négatif, mais il cherche à représenter comment le langage marche en général. Et puisqu'il représente ça comme ça, il essaie de trouver une représentation de la langue, sans à le rendre nulle partage. Mon morceau le plus important. on a l'idée de modèle. Alors, c'est un modèle qui s'appelle un transformer, c'est une architecture assez particulière. On va y aller très rapidement dans le prochain slide, mais ça requiert un entraînement spécifique et ça on ne voit pas. Et enfin large, pourquoi ? Parce que vous vous souvenez, on peut enlever la dernière couche et rajouter un autre truc à la place, et ça permet de donner des très bons modèles qui ont déjà été entraînés pour vous. Et donc vous avez déjà une certaine connaissance qui est encodée dans ces modèles, une connaissance de la langue. Ils sont complexes, ils sont puissants, mais ils sont très lourds. Alors, pour pouvoir parler de tout ça, il y a 2-3 notions qui sont assez importantes à avoir. La première c'est la notion de tokenisation. Puisque tout à l'heure vous parle de maths, mais le texte est math au fond. On va transformer le texte en une suite de chiffres. Et ça, ça se fait assez facilement, en fait, assez basiquement. On commande chaque token, vous imaginez que c'est un mot, on enlève les mains, et on lui donne un code. Et ce code, c'est un chiffre. Alors, la plupart du temps, on crée de très très large vocabulaire pour les réseaux, pour les LLM. Ça va de 1 à 350 000, je crois. Et en fait, à chaque mot, on associe un code. Et en fait, au lieu d'avoir une suite de texte, ça nous fait une suite de chiffre. Et en fait, quand on commence à faire des notes dessus à la fin, ça nous donne une représentation beaucoup plus lourde, mais beaucoup plus, il y a plus de sens pour la machine. ce processus de passer du texte à un chiffre, c'est de la tokenisation, et c'est un sine qua ner. Maintenant, on va essayer de suivre l'idée de transformeur, combien de jours on va y passer. Ce n'est pas une complexité, mais en fait, un transformeur, c'est un modèle qui a des règles, et qui fait deux choses. On rentre des entrées dans notre cuisine, et on en crée une représentation cachée, par exemple, le charmeur de voirins. Ça nous sort un dicteur, une suite de chiffre qui représente cette phrase. Et ensuite, on va la passer à un décodeur, donc un deuxième réseau, qui, à partir de cette suite de chiffre, recrée le

texte. Donc en fait, c'est un peu le jeu du faussaire. On crée un truc, on encode un truc, et une fois qu'on l'a encodé, on essaie de le décoder. Je pense que là, vous voyez déjà l'idée de traduction, quand même, qui commence à énerver. On transforme quelque chose, et on le retransforme dans un truc, etc. Et en fait, ça, c'est très puissant, parce que finalement, ce que ça fait, c'est que ça apprend une représentation cachée du langage qui est utile, qui contient les règles du langage, assez de règles et assez de connaissances, voire recréer le langage. Et donc, quand on voit que ça a été écrit en deux morceaux, ça donne un peu plus de sens au fait qu'il y a plusieurs LLM. Il n'y en a pas qu'un seul. chaque LLM a une famille qui permet de faire quelque chose en particulier. Vous connaissez le GPT par son emblème chat-gpt. En fait, c'est des modèles autoregressifs qui sont basés que sur un décodeur. Et leur idée, c'est de faire de la génération. Vous leur donnez un, deux ou trois trucs, une séquence de textes. Et en fait, leur but, c'est de continuer à générer des choses. Donc en fait, c'est vraiment un décodeur. Vous avez la famille BERT qui, eux, c'est que des encodeurs. En fait, ils créent une représentation à partir du texte initial. Donc, eux, ils sont très, très bons à faire de la classification. Ils sont très, très bons à faire du résumé de choses. Dans cette phrase, je suis très déçue par ce film, il apprend quand même quels mots sont positifs ou négatifs. C'est comme ça qu'on peut faire de la classification de textes. Vous imaginez la classification de sentiments, vous imaginez la classification de types de textes. Est-ce que c'est de la poésie ou du texte académique, quelque chose comme ça. Et enfin, le dernier, un peu moins connu, c'est le BERT ou le T5. C'est ce qu'on appelle du sequence to sequence. C'est des modèles de traduction. C'est ce que Google utilise dans Google Translate par exemple. Et donc, c'est surtout utilisé pour faire de la traduction, mais aussi pas mal de résumé de textes, puisque ça prend une séquence, celle en code et celle à des codes. Donc, il y a moyen d'apprendre à résumer les choses. Tous ces entraînements, toutes ces choses ont besoin de beaucoup d'entraînement. Et quand je vous dis non supervisé, on va ensemble voir pourquoi il n'y a pas besoin de l'appel. En fait, l'idée de GPT, parce que je pense que c'est intéressant pour tout le monde, c'est que quand on a une phrase, on entraîne, on lui demande un premier mot, puis un deuxième, puis un troisième, et on lui donne un mot pour prédire le prochain mot. Pour chaque mot dans son vocabulaire, pour chaque mot qu'il connaît, il va vous dire, je suis à peu près sûr que c'est ça qui suit. The boy went to work, il est à peu près sûr que c'est Playground qui passe. Mais, c'est là où on voit ça, il n'y en a pas qu'un mot possible, il y en a plein. Et c'est à nous de faire le choix de quel mot on va prendre, souvent parce que c'est celui qui a la plus force pour nous. Et on entraîne comme ça. On lui donne un bout de phrase, et puis on peut prendre du texte qui existe partout sur le net, simplement on découpe jusqu'au dernier mot, et puis on laisse continuer et essayer de trouver. S'il trouve que c'est Playground, c'était le bon mot, on le récompense, on ne l'envoie pas à la casse. Si ça ne le fait pas, il part au budget. C'est comme ça qu'on entraîne. C'est comme ça que GPT marche. Il apprend à prédire des trucs. Il apprend que le général, après vie, ça doit être un mot. Bert, par exemple, apprend une façon différente, quand même assez semblable, mais qui ne requiert pas non plus de données à noter. Simplement, au lieu de prendre la phrase, on masse un morceau. On ne lui demande pas de prédire le prochain mot, on lui demande juste de prédire un mot, étant donné le contexte. Et en fait, quand on le masse et qu'on essaie de prédire des choses, Bert va apprendre qu'en général, à cet endroit-là, par exemple, entre un verbe et un… Je ne sais pas ce que c'est que j'ai dit. Ça va être un mot. En gros, il va enquêter le type de mot tout seul. Je n'avais pas besoin de lui appeler les tags grammaticals parce qu'il va en voir tellement qu'il va apprendre à faire des références par lui-même. Ça se ressemble beaucoup, mais ce sont deux pages différentes, mais dans les deux coeurs, vous voyez, vous n'avez pas besoin de le persuer à noter des trucs et à passer très loin. Et c'est aussi ça qui fait leur puissance. Par exemple, on peut leur passer l'entièreté de l'Ukipédia. Automatiquement, on en met un mas, ça marche. Donc, et là, c'est un petit morceau anti-délinq, désolée, mais qu'est-ce qui se passe dans la vraie vie ? Vous avez besoin de faire du fine-tuning. Vous savez, votre LLM qui avait déjà entraîné, votre LLM des gens entraînés n'a pas été fait gratuitement. Il existe. Il y a des gens qui ont pris des millions de terres de gigas, de terres de données, de textes, beaucoup. L'entièreté de l'Ukipédia, l'entièreté du monde, et qui l'ont passée pour faire le LLM. Ça coûte très très cher. Ça prend plutôt longtemps. Ce qui fait que le groupe investe dans les modèles, dans l'opening high, dans le good girl. Ah oui, et puis le good girl lui fait des petites feintes, genre, il s'appelle good man maintenant, plus bon que le good, donc il ne sait pas s'il est même. Mais en gros, on est resté dans un écosystème qui est fait de six, sept gros vecteurs. Mais par contre, après, le truc vraiment bien, c'est que vous prenez l'optique de LLM, c'est-à-dire du fight tuning, ça prend beaucoup moins de temps, c'est beaucoup moins cher, et ça permet d'avoir des résultats, parce que le LLM a un petit peu plus d'angle pour tous ces corpus créatifs, et donc vous nous avez donné des petites règles un peu plus fines sur la page que vous avez. c'est vraiment pratique aussi quand on a, quand on veut la recherche. Donc, pour le LLM, qu'est-ce que c'est un modèle ? Un modèle qui est entraîné, c'est tout, mais attention, il y a une construction un sur l'autre, c'est-à-dire vous vous venez avec des choses qui existent déjà, donc il a déjà appris des choses, le modèle, notamment des biais, notamment, c'est comme ça qu'on se retrouve avec des modèles racistes, sexistes par exemple, si on a mal entraîné et si on ne sait pas d'où ça vient. Ensuite, il faut ajouter des données très adaptées à la tâche. Donc, en fonction de la tâche que vous avez déjà l'avoir définie, qu'est-ce que c'est un succès, qu'est-ce que c'est un échec dans la tâche, d'avoir créé les données pour ça, et donc de gérer dans l'adaptation à la tâche le biais des modèles précédents. Le but d'un modèle, c'est de répondre à la tâche sur laquelle vous entrenez. Il n'y a aucune garantie de véracité, il n'y a aucune garantie que votre modèle soit bon sur autre chose. Il est bon sur un truc, c'est tout ce qu'il est fait, et du coup, il n'y a aucune donnée qui est collée dans les déraines des règles. Si vous me disiez, oui, généralement, appréhender un verbe, c'est un cv ou un cwi, ce n'est pas le truc. Donc, vous n'avez aucune règle qui vous garantit que linguistiquement, ça va avoir du sens. Juste l'entraînement, vous lui faites confiance pour avoir appris ça suffisamment bien. Et donc, préconisation numéro 1, c'est de réfléchir avant d'acheter quand on a un modèle. Parce que le truc le plus important, c'est quelle tâche. On ne va pas juste dire, ah, je veux faire ça. Non. Quelle tâche ? Ça veut dire quoi réussir, ça veut dire quoi échouer ? C'est quoi les limites de la tâche que je fais ? Et du coup, quand je connais ma tâche et ce que je veux en faire, je sais quel modèle choisir de façon souveraincente. Je ne vais pas, si je veux faire de la classification de textes en espagnol, rendre Berthe 40 000 tokens en anglais. C'est bizarre. Ce qu'il m'aura pris sur l'anglais et pas sur l'espagnol. Si je veux pouvoir faire de la traduction, je ne vais pas prendre un modèle qui m'utilise. Des choses comme ça, à ce moment-là. Et enfin, une fois que j'ai choisi mon modèle consciemment, soit en ayant lu mon état de l'art, soit en ayant un contact avec quelqu'un dans ce domicile, une fois que j'ai bien défini ma tâche et que j'ai fait mon entraînement, maintenant, la question c'est, quels sont les biais potentiels inhérents à ma tâche et à moi, inhérents au modèle, et du coup, comment ça se conduit ? Et si ça on peut le faire, on peut corriger les biais. Par exemple, un modèle raciste, ce que fait la JPP, il le corrige un peu à la main en fait. Ils ont des règles en plus. Donc, on peut ensuite rajouter des choses par-dessus si les biais sont connus. Donc, si vous oubliez tout ce que vous avez dit avant, qu'est-ce que c'est que la JPP ? C'est un moteur de Pontiac, une voiture à pédale. On va très, très vite, très, très loin, mais il y a beaucoup de chances de rentrer dans un ordre et de perdre le contrôle. Parce que oui, quand on fait de la recherche applicative, c'est magique, parce que ça accélère, ça démocratise l'accès à la puissance de calcul, ça permet d'avoir la puissance de centaines d'heures d'entraînement avant, ça c'est bien. Ça permet d'avoir des tâches varier de la langue, on a la reconnaissance d'un cité nommé, on a la traduction, on a la classification de textes pour nommer quelques-uns et donc on peut avoir beaucoup de tâches variées. Il suffit d'avoir ces petites données à soi et toute la tâche nous est donnée. Et ensuite, tant qu'on a une tâche bien circonscrite, on arrive à avoir une adaptation qui est vraiment bien et qui marche très bien dans la plupart des cas. Par contre, LLM n'est pas un marteau doré. C'est-à-dire qu'il y a un travail de réflexion avant à dire, est-ce que c'est vraiment utile d'utiliser un LLM ou est-ce qu'il n'y a pas quelqu'un qui a déjà fait quelque chose d'utile ? Par exemple, je pense qu'il y a beaucoup de chercheurs en ce moment qui sont en train de faire de l'annotation linguistique en utilisant le chat GPB. Il y a des gens comme Donar Meuf, Instanza et Dupay qui ont déjà fait l'outil et que l'outil marche depuis des années. C'est utilisé beaucoup. Donc c'est un peu l'effet, si tout ce qu'on a c'est un marteau, tout ressemble un peu. Ce n'est pas forcément pertinent dans tous les domaines. Le deuxième c'est de se dire, est-ce qu'on peut vivre avec le poids des données de pré-entraînement. La plupart du temps, les gens disent oui. Mais si vous travaillez sur des données sensibles, des sujets sensibles, des choses comme ça, ça vaut alors beaucoup de se poser la question. On peut vite faire vraiment n'importe quoi. Parce que si on n'a pas bien posé sa question de recherche, si on n'a pas vérifié que le LLM c'était la meilleure solution possible pour nous aujourd'hui, on se retrouve pris dans un grenage et on fait n'importe quoi très vite. notamment mental. Il y a beaucoup d'ingénieurs qui font un LLM, ils font des trucs et puis à la fin leur disent mais ça sert à quoi ? Et ça pue vraiment. Donc on va vite dans la notion de performance en oubliant la notion des tâches initiales. Et enfin, et ça c'est un aspect personnel, je ne vais pas vous mentir, c'est ma paroisse, c'est que le pouvoir réside dans les mains de ceux qui ont la capacité à pouvoir entraîner. Donc deux fois pour sortir de ça, soit on accepte que c'est Google et Facebook et Meta qui vont nous donner et qui vont réussir à faire des choses, soit on essaie de rééquiper la recherche avec des modèles un petit peu plus légers, ce qui existe, ce qui se fait, on essaie de le faire et on essaie de le faire. Mais c'est vrai que ce n'est pas les plus grands. Et enfin, vraiment le dernier point dont j'ai besoin, que je m'en sens obligée de dire, c'est que ce sont des modèles de langues, pas des modèles de faits. Il n'y a aucune véracité dans ce que génère ChatGPT par exemple. Ils prennent à générer de la langue cohérente et en fait c'est un peu des politiciens. Ils vont dire n'importe quoi à partir du moment où ça vous convainc. Ce que ça sort, c'est de l'anglais, du français, de l'espagnol, du turc, cohérent, c'est bon en fait, c'est considéré comme étant bon. Donc en fait, le fait que souvent vous pouvez demander à ChatGPT donne-moi les dates de ceci, cela et qu'il répond correctement, en fait ce n'est pas son but, c'est juste qu'il répète les données d'entraînement, il a juste vu que le chute du mur de Berlin en 1989, ça allait ensemble la plupart du temps. Mais il n'y a aucune garantie que ça marche et ça n'est absolument pas ce qui est vrai. C'est extrêmement facile de forcer ChatGPT à vous dire des caméras. Vraiment, vraiment, c'est très facile. Moi, il va réussir à me dire des trucs que les modèles de langue, le large modèle de langue, ça a daté de 1956. Donc, encore une fois, si vous connaissez votre tâche et que vous savez que le LM, c'est potentiellement une bonne idée, oui, mais par contre, vous ne pouvez pas faire confiance quand c'est une autre tâche que ceux que je peux vous montrer. Merci beaucoup.