Comment ces IA inventent-elles des images ?

Les IA (intelligence artificielle) sont de plus en plus présentes dans notre vie quotidienne et ce Youtube vidéo explique comment elles peuvent être utilisées pour créer des images. Il montre comment les algorithmes sont capables d'utiliser des données pour générer des images originales et créatives. Cette vidéo est un excellent moyen d'en apprendre plus sur l'intelligence artificielle et sur la façon dont elle peut être utilisée pour créer des images.

Cette vidéo explique comment les intelligences artificielles peuvent générer des images à partir d'une description. Les algorithmes utilisés pour cela sont appelés des IA génératives et reposent sur l'apprentissage supervisé. Les IA génératives fonctionnent en prenant en entrée une série de nombres et en produisant une image en sortie. Cependant, cette méthode ne fonctionne pas bien car elle ne peut pas prendre en compte la part d'aléatoire et de synthèse d'information nécessaire à la production d'images. En 2014, une équipe de l'Université de Montréal a proposé une méthode plus efficace pour générer des images à partir d'une description.

Source : UCaNlbnghtwlsGF-KzAFThqA | Date : 2023-01-13 17:00:34 | Durée : 00:22:52

→□ Accéder à CHAT GPT en cliquant

dessus

Voici la transcription de cette vidéo :

Depuis maintenant plusieurs mois, vous n'avez certainement pas échappé à la déferlante des images qui sont générées par des intelligences artificielles, des IA. Qu'il s'agisse de DALL·E, de Midjourney ou encore de Stable Diffusion, il existe maintenant plusieurs programmes extrêmement efficaces, capables de créer de toute pièce une image à partir d'une description.
Et ce que font ces IA, ça n'est pas du Google Image. Elles ne vont pas chercher une image existante, mais elles produisent des images nouvelles, qui n'existent pas, et qu'elle créent à la demande. Il peut s'agit de photographies, de peinture ou d'art numérique,

Et on peut même parfois spécifier le type d'objectif photo ou le style du peintre que l'on souhaite retrouver. On appelle ces types d'algorithmes des IA génératives, car elles sont capables de créer du contenu à la demande, en intégrant à la conception une part d'aléatoire,

De façon à pouvoir obtenir plein de variations à partir de la même demande. Si vous avez essayé ces IA génératives, vous avez peut-être été comme moi ébahis par leurs capacités à faire quelque chose qui semblait impossible il y a seulement quelques

Années. Ca parait presque magique tellement ça fonctionne bien. Mais au fait, comment ça fonctionne vraiment ? Eh bien c'est ce qu'on va voir ensemble aujourd'hui ! [jingle] Commençons par quelques remarques de base sur la terminologie. Ce qu'on appelle l'intelligence

Artificielle, c'est un domaine aux contours un peu flous. On peut dire que son objectif est de réaliser à l'aide de

machines, des tâches traditionnellement réalisées par des humains, et dont on estime qu'elle requièrent de l'intelligence. Sans trop préciser le sens qu'on donne à ce terme.

[DIAGRAMME Au sein de l'IA, il y a un champ de recherche assez bien identifié qu'on appelle le machine learning, ou apprentissage automatique. L'idée est de s'attaquer notamment à des questions d'IA, en mettant au point des algorithmes capables d'apprendre. Et qui réalisent leur apprentissage à partir de données auxquelles ils ont accès.

Et dans le machine learning, il y a le deep learning. Et ça, c'est l'idée d'utiliser une classe assez spécifique d'algorithmes que sont les réseaux de neurones profonds, et qui ont connu un énorme succès depuis une dizaine d'années.] Aujourd'hui quand on utilise le terme IA, il s'agit presque systématiquement de deep

Learning, tant cette approche est efficace. Je ne vais pas évoquer ici les principes de base du deep learning, j'avais fait toute une vidéo sur le sujet il y a quelques temps, je vous invite à aller la voir si ça vous intéresse.

Quand on parle d'algorithmes capables d'apprendre, cela fonctionne souvent selon le même principe. Imaginez qu'on veuille créer un algorithme qui soit capable de reconnaitre ce qu'il y a sur une image. [ALGO On veut un programme à qui on donne une image en entrée, et qui produise un mot

En sortie. Et on veut que cet algorithme réponde « Chat » sur cette image, sur celle-ci « Voiture », etc. Pour réaliser cela, on va utiliser ce qu'on appelle de l'apprentissage supervisé. Au départ notre algorithme va être ignorant, et répondre n'importe quoi quand on lui

Présente une image. Et puis on va lui fournir une grande base de données d'exemples. C'est-à-dire des milliers d'images de chats, de voiture, de tasses, avec à chaque fois la réponse

qu’on attendrait de lui.] En utilisant ces exemples, l’algorithme va peu à peu ajuster ses paramètres pour

S’améliorer, il va apprendre. Jusqu’à être capable de réaliser lui-même le travail de façon suffisamment correcte. On dit alors que l’algorithme aura été entrainé, et on pourra commencer à l’utiliser. Cette façon de faire, c’est donc l’apprentissage supervisé. De façon plus générale si on

Veut qu’un algorithme prenne en entrée X et réponde Y, on lui donne une base de données d’exemples (x,y), et on l’entraine en espèrant qu’il va finir par savoir faire le lien. C’est ce qu’on appelle parfois une tâche de prédiction, puisqu’on donne X en entrée

Et on demande à l’algorithme de prédire Y. Dans mon premier exemple X ce sont des images et Y ce sont des mots. On classifie les images avec des mots. Mais le principe fonctionne avec plein d’autres concepts. Par exemple X ce peut-être toutes les caractéristiques d’une transaction bancaire, et Y est une

Classification pour estimer si la transaction est frauduleuse ou pas. Ou bien X c’est le texte d’un e-mail et Y c’est une classification entre Spam/Non-spam. Beaucoup des applications actuelles concrètes du machine learning reposent sur ce principe d’apprentissage supervisé, qui permet de réaliser des tâches de prédiction.

Ca semble un peu magique présenté comme ça, de penser qu’un algorithme peut reconnaitre des images et associer les bons mots, mais il faut bien voir que tout ce que fait l’algorithme, c’est de manipuler des nombres. Tout se fait via des opérations mathématiques.

[CHIFFRES Par exemple sur la classification des images, voici comment on fait en pratique. On définit d’abord un nombre fini de catégories : « Chat », « Voiture », « Tasse », etc. Et on leur attribue des numéros : 1,2,3 etc. On peut faire disons

1000 catégories,

Et à chaque image de notre base de données on va associer une de ces 1000 catégories. Et pour l'image elle-même, on va aussi la représenter comme une série de nombres. Une image 600×600, ce sont 360 000 pixels, chacun de ces pixels va être codé en rouge/vert/bleu

Donc il faut 3 nombres par pixel. Au total je peux décrire complètement mon image en un peu plus d'un million de nombres. Donc ce que je cherche à faire, c'est un algorithme qui prenne en entrée une série d'un million de nombres qui représentent l'image, et qui me crache en sortie un nombre

Entre 1 et 1000 représentant la catégorie qu'il pense être la bonne.] On n'a pas besoin d'expliquer à l'algorithme ce qu'est une image, un chat ou une voiture; lui, il manipule juste des nombres. Et il va s'en faire une idée tout seul en essayant

De mettre en relation les valeurs des pixels des images et les 1000 catégories qu'on aura prédéfinies. Ces algorithmes de classification d'images fonctionnent maintenant extrêmement bien, avec des performances proches de celles des humains. Alors intuitivement, on se dit que ça, ça n'a pas l'air très très éloigné de ce qu'on voudrait faire pour générer

Des images. [REVERSE Pour classifier des images existantes, en entrée X on a des images, et en sortie Y un mot qui les décrit, et une base de données pour réaliser l'entrainement. On n'a qu'à entrainer un algorithme du même genre sur la même base de données, mais à l'envers.

On lui donne un mot en entrée, et on attend qu'il produise une image en sortie. Facile, non ?] Eh bien non, ça ne marche pas comme ça. Il y a plusieurs raisons à cela. Dans une tâche de prédiction, comme la classification des images, on donne en entrée beaucoup d'informations,

Des centaines de milliers de pixels, et en sortie on récupère

quelque chose de très limité, juste un numéro de catégorie par exemple. L'algorithme de classification réalise en quelque sorte de la synthèse d'information. Ici c'est l'inverse qu'on veut, on donne très peu de choses en entrée, disons juste

Un mot. Et on s'attend en sortie à récupérer une image entière, qui contient donc beaucoup plus d'information que ce qu'on a mis au départ. Ca ne peut pas marcher avec un simple apprentissage supervisé. En fait si, on peut essayer. Mais ce qu'il risque de se passer c'est que notre algorithme,

Une fois entrainé, chaque fois qu'on lui demande une image de chat il va nous ressortir exactement l'image de chat qu'il avait dans sa base de données. Ou bien une sorte de moyenne de toutes les images de chat qu'on lui a fournies. Et c'est pas ça qu'on veut.

L'autre aspect important, c'est qu'on veut que notre algorithme de génération d'image ait une part d'aléatoire. Si je lui demande 50 fois un chat, je veux qu'il me produise 50 images différentes, pas 50 fois le même chat. Générer ça n'est pas la même chose que prédire. Donc vous voyez que pour créer

Des images à partir d'un texte, prendre une approche traditionnelle de prédiction par apprentissage supervisé, eh bien ça ne va pas marcher. Il faut faire autrement. Alors voyons comment. [jingle] En 2014, une première façon de faire assez efficace a été proposée par une équipe

De l'Université de Montréal : les réseaux adversariaux génératifs, ou GAN en anglais. Alors qu'est-ce que ça veut dire ce truc. Pour voir ça, prenons un problème plus simple que celui que j'ai évoqué. Imaginons pour commencer que je veuille uniquement créer des images de chat.

[GAN Je dispose d'une grande base de données d'images de chat, je veux juste être capable d'en créer plein d'autres, de nouvelles, qui n'existent pas initialement dans la base, et qui ressemblent bien à des chats. Pour cela, je vais entrainer

non pas un, mais deux réseaux de neurones.

Le premier, on va l'appeler le faussaire. Il aura pour rôle de générer aléatoirement des images à partir d'une série de paramètres. Et le deuxième, on va l'appeler l'expert, et sa tâche va être simple : si on lui présente une image, il va devoir apprendre à reconnaitre

Si l'image est une vraie image de chat, qui provient de la base, ou bien une image inventée. Rappelez-vous le principe de l'apprentissage, au début ces algorithmes sont mauvais. Le faussaire, quand on lui demande de produire une image de chat, il va produire des images

De ce genre, du bruit complet. Et l'expert, quand on lui présente deux images en lui demandant quelle est la « vraie », au début il va répondre au pif. Et puis à force d'essais et d'erreurs, ces réseaux vont progresser car à chaque

Fois on leur donne le résultat. On indique au faussaire s'il a réussi à berner l'expert. Et l'expert lui sait s'il a correctement discriminé les images qu'on lui présente. Au départ, l'expert va assez vite apprendre à reconnaitre les vrais chats des fausses

Images, qui sont initialement assez pourries. Et puis peu à peu le faussaire va progresser et faire des images qui seront de mieux en mieux, et ça va forcer l'expert à améliorer son niveau.] Et en laissant les deux s'entrainer ensemble suffisamment longtemps, on finira si tout

Va bien par avoir un faussaire capable de produire des images très semblables aux vraies, et un expert qui aura bien du mal à faire la différence. Ce principe d'entrainement, on l'appelle « adversarial », puisque les deux réseaux de neurones s'affrontent et progressent de concert. C'est comme si vous demandiez à deux joueurs de foot de

S'entrainer ensemble : l'un fait le tireur de pénalty et

l’autre le gardien, avec l’idée qu’ils vont conjointement progresser. Ces réseaux adversariaux génératifs, ou GAN, existent maintenant depuis plusieurs années, et vous les avez certainement vus à l’oeuvre si vous connaissez le site This Person Does Not Exist.

[THIS PERSON Sur ce site, on a des réseaux ont été entrainé à partir de photos de visages, et qui sont donc excellent à inventer de nouveaux visages de toute pièce. Il y a parfois des défauts mineurs mais dans l’ensemble c’est assez bluffant.

Dans le même genre, vous avez aussi pour les voitures This Car Does Not Exist, pour les maisons This House Does Not Exist, et ah oui, bien évidemment : This Cat Does Not Exist.] Ce type d’algorithme donc est très efficace pour produire des images d’un type donné

Et fixé à l’avance. Si l’entrainez sur des chats, il vous fera des chats. Mais vous voyez qu’on n’est pas encore au stade où on peut rentrer librement du texte et avoir une image qui mélange plusieurs concepts. On va y venir, mais avant ça, prenons un peu de recul et essayons de comprendre la

Différence entre ce que font les réseaux génératifs, et l’apprentissage supervisé qu’on a vu avant, celui qui permet par exemple de classifier des images. [COMPARAISON Dans l’apprentissage supervisé, on donne X en entrée, et on attend que l’algorithme

Nous prédise Y en sortie, et pour ça on lui donne plein d’exemples de X et de Y. Avec les réseaux génératifs, on donne plein de X en entrée, par exemple des images de chat, et on dit à l’algorithme « Vas-y donne m’en d’autres ».]

[2D Essayons de se représenter la différence en imaginant qu’il n’y ait que deux dimensions. Par exemple si X et Y sont chacun de simples nombres réels. Dans ce cas là on peut représenter chaque exemple (x,y) de la base de données comme un point dans un diagramme en 2D.

Dans l’apprentissage supervisé, on veut que l’algorithme puisse prédire Y à partir de X. Donc en gros on veut que notre algorithme arrive à faire une sorte de régression, qui lui permette de deviner à peu près la valeur de Y si on lui présente une nouvelle

Valeur de X qu’il n’a jamais vu dans la base. Ok maintenant comment ça se passe pour un algorithme génératif, à qui on demande de construire de nouveaux exemples à partir d’une liste de X. Imaginez que dans mon cas X soit en 2 dimensions, donc décrit par deux nombres. Deux c’est pas beaucoup, tout

À l’heure une image on a dit que c’était 1 million valeurs, mais nous on va en regarder que deux. Appelons les x1 et x2. Donc ma base d’exemple c’est une liste de valeurs (x1,x2). Imaginez que je représente chaque exemple dans un diagramme et que j’obtienne ça. Ou bien ça. Manifestement il se passe

Quelque chose. Dans mon premier cas on voit que tous les exemples sont groupés dans un certain coin. Dans mon second cas, on voit qu’il y a comme une structure sous-jacente. Si on me présentait une base de données de ce genre, je serai capable de dire quelque

Chose d’intéressant sur la façon dont sont répartis les exemples, sur leur distribution. Ils ne sont pas quelconques, il ne sont qu’une petite partie de l’espace total des possibilités. Et avec les images de chat c’est pareil. Ok il y a 1 million de dimensions donc c’est

Dur à visualiser, mais les images de chat sont ne sont pas des images quelconques. Dans l’espace immense de toutes les images possibles et imaginables, les images de chat ne sont qu’une minuscule région. Et ce qu’on veut faire, c’est en savoir plus sur cette région.

Car si on arrive à dire un truc intelligent sur la manière dont sont distribués les exemples d’une base de données, on sera capable de faire de l’échantillonnage. C’est à dire de

produire de nouveaux points qui ont de bonnes chances d'être du même genre.

Dans mes exemples simples en deux dimensions, si je considère par exemple ces points-ci. Ou ceux-là dans le diagramme d'à côté. Ce sont des points qui n'étaient pas dans les bases initiales, mais dont on a bien envie de parier qu'ils auraient pu y figurer, qu'ils ont certainement les mêmes caractéristiques qui nous intéressent.

Et pareil pour mes chats, je voudrais être capable de créer une image qui soit nouvelle, et que j'aille la prendre dans la sous-région qui contient les images de chat.] Et vous voyez que cette gymnastique est différente de l'apprentissage supervisé. On ne fait

Pas de la prédiction au sens on me donne X et je prédis Y. Non, ici, on a ce qu'on appelle de l'apprentissage non-supervisé. L'apprentissage non-supervisé, c'est quand on a des données en entrée qui ne sont pas séparées en X et Y, ce sont juste des exemples X, et qu'on essaye de comprendre

Comment ces X sont distribués dans l'espace de toutes les possibilités. De façon à pouvoir, par exemple, en générer de nouveaux par échantillonnage. Evidemment à nouveau d'une part ça ne se passe pas dans un espace à deux dimensions, et d'autre part les régions qu'on cherche à identifier n'ont pas des formes simples.

Ca se voit très bien par exemple par le fait que la moyenne de deux images de chat ne donne pas une bonne image de chat. Mais si vous arrivez à correctement caractériser la région dans laquelle se trouvent toutes les images de chat de votre base initiale, vous pouvez avec une certaine confiance échantillonner

De nouvelles images en allant chercher au même endroit. Bien, donc générer de nouvelles images, ça n'est pas une tâche d'apprentissage supervisé mais c'est une tâche d'apprentissage

non-supervisé. Et on l’a dit, les réseaux adversariaux GAN marchent bien pour ça, mais ne fonctionnent que si on donne plein d’exemples

En entrée. Et il faut le faire pour chaque type de catégorie qu’on veut générer : des chats, des voitures, des appartements, etc. Les applications sont donc limitées, et cela explique que l’on n’ait pas vu de choses très spectaculaire depuis This Person Does Not Exist. Sauf depuis l’été 2022 avec

L’apparition de DALL·E et Stable Diffusion, qui n’utilisent pas les GAN. Et donc voyons maintenant cette nouvelle technique qui fait fureur en ce moment : les algorithmes de diffusion. [jingle] Avant de voir comment les algorithmes de diffusion sont capables de créer des images en prenant

Un texte libre comme point de départ, voyons déjà comment ils fonctionnent sur la même tâche que celle des GAN : générer plus d’images d’un type donné. Le principe de base est assez simple, et ça semble presque miraculeux que cela fonctionne.

[BRUIT Imaginez une image de chat. Je peux essayer de dégrader cette image en la bruitant. C’est facile, je génère une image qui est du pur bruit, c’est-à-dire que chaque pixel a une couleur totalement aléatoire. Et puis je mélange les deux images avec certaines proportions. Disons 95% de chat et 5% de bruit.

J’obtiens une image de chat légèrement dégradée. Et ce que je vais faire, c’est d’abord une étape d’apprentissage supervisé : je vais entrainer un réseau de neurones à débruiter les images. C’est à dire que je crée une base de données avec en entrée des images bruitées et en

Sortie les images d’origine. Et je donne ça à mon réseau pour lui apprendre à enlever le bruit. Ici il doit enlever 5% de bruit. Et ce qui est intéressant, c’est que je peux faire ça avec différents niveaux de

Bruit. Par exemple je peux bruiter une image à 45% et à 50%, et entrainer mon réseau à retirer 5% de bruit pour passer de 50 à 45. Et pareil avec 90 et 95%. Donc si j'ai une base de données d'images de chat, je peux toutes les bruiter à plein

De niveaux différent niveaux et entrainer mon algorithme avec. Et à la fin de son entrainement, mon algorithme sera devenu super fort à débruiter des images de chat.] Maintenant une fois que mon algorithme de débruitage est entrainé, imaginez que je

Lui donne une image qui soit 100% de bruit, j'ai tiré chaque pixel au hasard. Et que je lui dise : vas-y débruite là progressivement. Retire 5% de bruit, puis encore 5%, puis encore 5%. Qu'est-ce qu'il va se passer ? [DEBRUITAGE Eh bien comme mon algorithme aura été uniquement entrainé à débruiter des

Images de chats, il va progressivement m'inventer une image de chat à partir du bruit initial. Et cette image ne sera aucune des images de la base initiale, ce sera une nouvelle image. Il va en quelque sorte « halluciner » une image de chat à partir du pur bruit qu'on lui donne en entrée.

Et en faisant ça, notre algorithme fait de l'échantillonnage parmi la distribution de toutes les images de chat possibles et imaginables. On tire un bruit au hasard, et il nous fait une image de chat avec. Cet échantillonnage est donc une forme d'apprentissage non-supervisé de la distribution des images de chat.]

Cette idée de génération à partir d'un débruitage progressif, c'est ça qu'on appelle les méthodes de diffusion. La technique a été proposée il y a quelques années déjà, mais elle vient de connaitre une accélération grâce à de nombreuses améliorations récentes.

A ce stade de ma description, ce type d'approche ne fait pas mieux que les GAN. Si vous faites ça avec des images de

voiture au lieu d'images de chat, vous allez entrainer un réseau de débruitage qui aura la capacité à halluciner des images de voiture. Donc on peut génèrer

Des images d'un certain type à condition d'en avoir donné suffisamment d'exemples en entrée, et de refaire l'entrainement. Mais il est possible d'étendre le concept pour le rendre plus polyvalent. Vous vous souvenez du problème de classification d'images : on a une grande base de données d'images,

Et chacune est associée à une classe : chat, voiture, visage, etc. Ce qu'on peut faire, c'est traiter tout d'un coup. [CONDITIONNEMENT Pour ça on va bruiter ces images et les donner à notre algorithme pour qu'il apprenne à les débruiter, mais en lui spécifiant à chaque fois la catégorie à laquelle appartient chaque image.

L'algorithme de débruitage sera ainsi capable de s'ajuster en fonction de la catégorie. C'est un principe qu'on appelle le conditionnement. On va conditionner notre débruitage par la catégorie de l'image. Et si on fait ça avec une base de données suffisamment grande, une fois l'entrainement

Terminé on aura un algorithme capable d'halluciner une image en étant conditionné par une classe. Si on lui dit voiture il va halluciner une voiture, et chat il va halluciner un chat.] Alors c'est bien, mais ça n'est pas encore tout à fait ce qu'on veut. Ici on est limités

Par les catégories qui ont été choisies au départ pour classifier les images. Or vous voyez que dans DALLE ou Stable Diffusion, on peut entrer librement du texte, faire des phrases, des descriptions. Comment on procède ? Eh bien il va falloir faire appel à certains algorithmes de traitement du langage naturel,

Comme ceux que j'évoquais dans ma vidéo sur le sujet. Et notamment les algorithmes permettant de faire ce qu'on appelle

de du plongement, embedding en anglais. [EMBEDDING L'idée du plongement, c'est de partir d'un mot ou d'un phrase, et de lui associer une série de nombres qui en quelque sorte représente numériquement

Son sens, sa signification. Mais plongée dans un espace mathématique abstrait. De façon à ce que deux phrases proches aient des plongements qui soient proches.] Il existe des réseaux de neurones qui peuvent réaliser ces plongements, et si on les utilise avec un algorithme de diffusion, on peut alors utiliser le plongement pour conditionner le

Débruitage. [CONDITIONNEMENT Ca marche comme tout à l'heure, mais au lieu d'être conditionné par une catégorie d'image, notre processus de débruitage sera conditionné par le plongement de la description de l'image. En pratique, on prend des toujours des images mais cette fois accompagnées d'une description

Libre, par exemple une légende. On calcule le plongement de la légende, et on entraine notre réseau à débruiter ces images, tout en étant conditionné par le plongement de la légende. Et si on fait ça avec suffisamment d'images légendées, eh bien on obtient un algorithme de génération d'image à partir d'une description.

Vous tapez un texte, l'algorithme calcule son plongement. Il génère ensuite une image 100% bruitée et ensuite va la débruiter progressivement, tout en étant conditionné par le plongement de la description qu'on a donné. Et si tout va bien l'algorithme va halluciner une image en prenant comme contexte la description qu'on a donné.]

Voici dans les grandes lignes comment fonctionnent les algorithmes de diffusion comme Stable Diffusion ou encore DALLE. Bon il y a encore pas mal de subtilités mais je pense que la vidéo est déjà bien assez lourde, et je vous renvoie au billet de blog qui accompagne

La vidéo si vous voulez plus de détails techniques. Avant de conclure, il y a un point que je dois quand même évoquer, ce sont les problèmes éthiques et juridiques posés par ces algorithmes. Il y a bien sûr la possibilité qu'ils offrent de créer de fausses images ayant l'air complètement plausibles, donc potentiellement

De faire de la désinformation avec. Il y aussi les question de propriété intellectuelle, puisque ces algorithmes ont été entrainés à partir d'images trouvées sur Internet, et qui ne sont pas forcément libres de droits. C'est particulièrement criant pour les dessins ou l'art numérique en général,

Puisqu'on peut demander à l'algorithme d'imiter le style d'un artiste sans que ce dernier ait eu son mot à dire. Ces questions sont plus complexes qu'il n'y parait, et je n'ai pas vraiment les compétences artistiques ou juridiques pour les traiter correctement. D'autant que les

Choses évoluent très vite et qu'il y aura certainement des effets de jurisprudence. Donc gardez à l'esprit ces interrogations car on risque d'entendre parler pas mal prochainement. Voilà c'est tout pour aujourd'hui, merci d'avoir suivi cette vidéo. N'hésitez pas à aller voir mes vidéos connexes sur le deep learning ou le traitement du langage.

Abonnez-vous si ça n'est pas déjà le cas et n'hésitez pas à venir vérifier régulièrement les nouvelles vidéos, même si Youtube ne vous les propose plus. Pour celles et ceux que ça intéresse, j'ai aussi créé récemment un serveur Discord communautaire, n'hésitez pas à le rejoindre pour venir discuter. Le lien est en description.

Et moi je vous dit à très vite pour de nouvelles vidéos. A bientôt.