

Groupe 1 - Post 3

Article : Vidéo "But what *is* a Neural Network?" 3Blue1Brown (YouTube)

Lien : https://www.youtube.com/watch?v=aircAruvnKk&index=2&list=PLZHQObOWTQDNU6R1_67000Dx_ZCJB-3pi&t=0s

Objectif : Comprendre comment un ordinateur arrive à reconnaître une image

Il est intéressant de comprendre la base de la reconnaissance d'image pour appréhender les enjeux techniques mais aussi toutes les possibilités offertes par cette technologie.

Cette vidéo de vulgarisation tourne autour d'un seul exemple: Comment trouver un algorithme qui, recevant une image d'entrée d'un chiffre manuscrit, retourne le chiffre en question? Il faut s'intéresser en premier au fonctionnement de notre cerveau lors de l'analyse d'une image: comment nous permet-il de dire que deux chiffres sont identiques, même si leur style d'écriture les rend graphiquement très différents.

Le but est de décomposer chaque chiffre par des constituants élémentaires, rond ou barres (par exemple un rond en haut suivi d'une barre verticale donne un "9") puis encore en sous-sections pour se focaliser que sur de petites parties du nombre à la fois. Chaque partie découpée, par analogie avec notre cerveau, correspond à un neurone que l'on organise par couche de la plus élémentaire (couleur de chaque pixel reçu) à la plus élaborée (chiffre entre 1 et 9). Chaque neurone contient une valeur entre 0 et 1 qui symbolise son degré d'activation et une fonction mathématique qui décrit son impact sur la couche suivante. Une combinaison de neurones actifs réunis détermine l'état des neurones de la couche supérieure.

La première étape est de scanner l'image à la recherche de ces lignes élémentaires. Pour cela, on définit une zone d'intérêt que l'on déplace sur la grille de pixels reçus. En additionnant les valeurs correspondant à la couleur de chaque pixel dans cette zone, on a une idée globale de quel constituant élémentaire de ligne ou de courbe se trouve à cet endroit (Grande valeur de la somme) ou s'il n'y a rien (la somme est nulle). Puis, à l'aide de ces informations, on reconstruit le petits bouts d'image de plus en plus grand en remontant les couches de neurones. Cette action est condensée par des multiplications matricielles qui prennent en compte l'effet de chaque neurone, et est facilement automatisée par l'ordinateur. Au final, l'ordinateur a plusieurs choix de nombres probables dans la couche la plus haute, il donne celui avec la plus grande probabilité à l'utilisateur.

Dans un exemple simple comme celui-ci, qui ne prend en entrée qu'une image noir et blanc de 28x28 pixels, il faut plus de 13'000 neurones pour son fonctionnement. Il est impossible de régler toutes les valeurs et leurs impacts entre elles à la main, d'où l'utilité d'entraîner les machines avec des exemples à grande échelle pour qu'elles s'auto-ajustent afin de maximiser leur performance: c'est le principe du deep learning.

Pour une application de reconnaissance faciale, la tâche s'avère bien plus complexe: les différence de traits du visages entre les individus sont beaucoup plus fines que la forme d'un chiffre, il faut donc beaucoup d'images pour entraîner l'algorithme à avoir un taux de réussite acceptable, les images doivent être de bonne qualité et en couleur ce qui fait exploser le nombre d'informations à traiter et donc de neurones présents dans le système.

Dans le prochain article, nous aborderons plus en détails les problèmes de fiabilité de l'algorithme: comment peut-on modifier une image de façon indétectables par un humain et

induire la machine en erreur, et quelles sont les répercussions de ces erreurs sur les individus.