

الجمهورية الجزائرية الديمقراطية الشعبية
وزارة التعليم العالي والبحث العلمي



جامعة سعيدة د. مولاي الطاهر
كلية التكنولوجيا
قسم: الإعلام الآلي

Mémoire de Master

**Spécialité : Modélisation Informatique des Connaissances
et du Raisonnement**

Thème

deep learning et apprentissage par
renforcement pour la détection du covid-19
sur des images IRM

Présenté par :

- DAHMANE Abdelkader
- DAOUDI Yahia

Dirigé par :

Dr.Djelloul Mokadem



Promotion 2021 - 2022

Remerciements

En premier nous aimerons remercier le bon dieu le tout puissant de nous avoir donné le courage et la volonté de réaliser ce projet. Nos sincères gratitudes à notre encadreur Mr Djelloul Mokadem pour leur disponibilité et leur conseil ainsi que leurs remarques qui nous ont permis d'améliorer la qualité de ce travail

On remercie aussi Mr Bouarara hadj ahmed pour son aide et son encouragement.

Nos vifs remerciements s'adressent également à tous nos enseignants de département informatique pour la formation qu'ils ont eu le soin de nous apporter le long de notre cursus universitaire.

DAHMANE, DAOUDI.

Dédicace

*Je dédie ce modeste travail à chère mère et cher
père qui m'ont soutenu et encouragé dans les
moments les plus difficiles...*

*À ma femme et mes enfants, mes chers frères et sœurs pour leur
patience et leur affection...*

À toute ma famille et à tous mes amis...

*À tous mes collègues d'étude surtout les étudiants de
2^{ème} année Master MICR promotion 2021-2022.*

*Je le dédie aussi à tous mes enseignants durant les
années des études*

*Et surtout les enseignants d'informatique
d'université de Saida.*

DAOUDI Yahia

Dédicace

*De ma part, et avec des grands sentiments et d'une joie
Immense que je dédie ce travail à mes très chers parents
qui ont sacrifié pour que je réussisse (Que Dieu ait pitié d'eux avec
sa grande miséricorde), afin que je puisse à mon tour vous
combler à ma femme et mes enfants , mes sœurs et frères vous
occupez une place particulière dans mon cœur. je vous dédie ce
travail en vous souhaitant un avenir radieux, plein de
bonheur et de succès à notre amis proches ZADI Salah
BAKKADOUR Abdelmadjid et HADEF Boubaker qu'on l'aime et
Merci pour votre grand soutien.*

DAHMANE Abdelkader

Résumé :

Le virus Corona est une des maladies qui ont causé des pertes humaines ces derniers temps c'est ce qui nous a poussés à faire ce travail pour le détecter de manière précise et rapide en utilisant des images par résonance magnétique(IRM), un apprentissage profond et l'apprentissage par renforcement.

- Le Deep Learning ou « apprentissage profond » est une famille d'algorithmes d'apprentissage (machine learning) pour entraîner des réseaux de neurones composés de plusieurs couches internes et potentiellement un grand nombre. Ces techniques ont permis des progrès importants et rapides dans les domaines de l'analyse du signal sonore ou visuel et notamment de la reconnaissance faciale, de la reconnaissance vocale, de la vision par ordinateur, du traitement automatisé du langage, traitement et détection des images.[1]

-L'apprentissage par renforcement permet à un agent d'apprendre un comportement qui n'a jamais été préalablement défini par l'homme. L'agent découvre l'environnement et les différentes conséquences de ses actions à travers des interactions avec celui-ci : il apprend de sa propre expérience, sans avoir de connaissances préétablies des buts ni des effets de ses actions. Ce travail s'intéresse à la façon dont l'apprentissage profond peut aider l'apprentissage par renforcement à gérer des espaces continus et des environnements ayant de nombreux degrés de liberté dans l'optique de résoudre des problèmes plus proches de la réalité. En effet, les réseaux de neurones ont une bonne capacité de mise à l'échelle et un large pouvoir de représentation. Ils rendent possible l'approximation de fonctions sur un espace continu et permettent de s'inscrire dans une approche développementale nécessitant peu de connaissances a priori sur le domaine. [2]

MOTS-CLÉS : virus Corona, détecter, images par résonance magnétique, apprentissage profond , l'apprentissage par renforcement, Deep Learning, réseaux de neurones.

Abstract :

The Corona virus is one of the diseases that have caused human loss in recent times that is what prompted us to do this work to detect it in an accurate and rapid way using magnetic resonance imaging, deep learning and learning by reinforcement.

- Deep Learning or "deep learning" is a family of learning algorithms (machine learning) to train neural networks composed of several internal layers and potentially a large number. These techniques have enabled significant and rapid progress in the fields of sound or visual signal analysis and in particular facial recognition, voice recognition, computer vision, automated language processing, image processing and detection. .

-Reinforcement learning allows an agent to learn behavior that has never been previously defined by humans. The agent discovers the environment and the different consequences of his actions through interactions with it: he learns from his own experience, without having pre-established knowledge of the goals or the effects of his actions. This work is interested in how deep learning can help reinforcement learning to manage continuous spaces and environments with many degrees of freedom in order to solve problems closer to reality. Indeed, neural networks have a good capacity for scaling and a large power of representation. They make possible the approximation of functions on a continuous space and make it possible to be part of a developmental approach requiring little a priori knowledge on the domain.

KEYWORDS: Corona virus, detect magnetic resonance imaging, deep learning, reinforcement learning, neural networks.

ملخص

يعد فيروس كورونا من الأمراض التي تسببت في خسائر بشرية في الآونة الأخيرة وهذا ما دفعنا للقيام بهذا العمل للكشف عنه بشكل دقيق وسريع باستخدام التصوير بالرنين المغناطيسي والتعلم العميق والتعلم عن طريق التعزيز.

- التعلم العميق هي عائلة من خوارزميات التعلم (التعلم الآلي) لتدريب الشبكات العصبية المكونة من عدة طبقات داخلية وربما عدد كبير.

لقد مكنت هذه التقنيات من تحقيق تقدم كبير وسريع في مجالات تحليل الإشارات الصوتية أو المرئية وخاصة التعرف على الوجه والتعرف على الصوت ورؤية الكمبيوتر والمعالجة الآلية للغة ومعالجة الصور والكشف.

- يسمح التعلم التعزيزي للوكيل بتعلم سلوك لم يسبق أن حدده البشر من قبل. يكتشف الوكيل البيئة والعواقب المختلفة لأفعاله من خلال التفاعل معها: يتعلم من تجربته الخاصة ، دون أن يكون لديه معرفة مسبقة بالأهداف أو آثار أفعاله. يهتم هذا العمل بالكيفية التي يمكن أن يساعد بها التعلم العميق في تعزيز التعلم لإدارة المساحات والبيئات المستمرة بدرجات عديدة من الحرية من أجل حل المشكلات أقرب إلى الواقع.

في الواقع ، تتمتع الشبكات العصبية بقدرة جيدة على التوسع وقوة تمثيل كبيرة. إنها تجعل من الممكن تقريب الوظائف على مساحة مستمرة وتجعل من الممكن أن تكون جزءًا من نهج تنموي يتطلب القليل من المعرفة المسبقة في المجال.

الكلمات المفتاحية: فيروس كورونا ، الكشف ، التصوير بالرنين المغناطيسي ، التعلم العميق ، التعلم المعزز الشبكات العصبية .

Table de matière :

	Résumé	04
	Introduction générale	12
Chapitre 01	Coronavirus (covid-19)	15
	1. Introduction	16
	2. le coronavirus et le covid-19?	16
	2.1 définition	16
	2.2 Etiquette et forme	17
	2.3 les symptômes de la maladie	17
	2.3.1 symptôme les plus fréquents	17
	2.3.2 symptôme les moins fréquents	18
	2.3.3 symptôme graves	18
	2.4 les types de coronavirus	19
	2.5 Transmission	19
	2.6 taux de létalité par âge	20
	2.7 Mortalité	20
	2.8 Test	21
	2.8.1 Test de dépistage virologique	21
	2.8.2 Test logique	21
	2.9 examen d'imagerie thoracique	21
	3 Nouvelles technologies de diagnostic	22
	3.1 Prise en charge et diagnostic par télé-médecine	22
	3.2 Diagnostique par intelligence Artificielle	22
	4 Traitement	22
	5 Vaccin	23
	6 Pandémie de covid-19 en Algérie	23
	6.1 Statistiques du début de la pandémie	24
	6.2 Historique	26
	6.2.1 Chronologique	26
	6.2.1.1 Première vague	26
	6.2.1.2 Deuxième vague	26
	6.2.1.3 Troisième vague	26
	7 Conclusion	27
Chapitre 02	Représentation des images	28
	1 Imagerie numérique	29
	1.1 définition de l'imagerie numérique	29
	1.2 Caractéristiques de l'image numérique	29
	1.3 comment un ordinateur voit une image	29
	1.3.1 image noire et blanc	30
	1.3.2 image niveau de gris	30
	1.3.3 image couleur	31
	1.4 le codage des couleurs sur 3 octets	32
	2 l'Image médicale	32
	2.1 Définition	32
	2.2 les différentes technologies d'imagerie médicale	32
	2.2.1 l'imagerie par résonance magnétique (IRM)	33
	2.2.2 Déroulement de l'IRM	33
	3 Relation entre covid-19 et image IRM	34
Chapitre 03	L'Apprentissage profond (deep learning) et l'apprentissage par renforcement (AL)	35
	1 introduction	36
	2 Définition	36
	3 Domaines d'application de l'apprentissage profond	37
	3.1 la reconnaissance faciale	37
	3.2 le traitement automatique de langage naturel	37
	3.3 voiture autonome	37
	3.4 Recherche vocale et assistants à commande vocale	37
	3.5 Traduction automatique	38
	3.6 Recherche automatique	38
	4 Représentation de quelle que types de réseau Neurone	38
	4.1 Neurone Formel	38
	4.2 Neurone multicouche	39
	4.3 Neurone récurrents	39
	4.4 Réseau de Hopfield	39
	4.5 Réseau Neurone convolutionnels	40
	5 Architecture de réseau de neurones profond	40
	5.1 les réseaux de neurones convolutifs	40
	5.1.1 les couches de réseau de neurones convolutionnels	41
	5.1.2 la couche de convolution	42
	5.1.3 la couche de Pooling	42
	5.1.4 la couche entièrement connectée (FC)	43

	5.1.5 la couche logistique ou softmax	44
	5.1.6 la couche de sortie (output layer)	44
	5.2 Réseau de neurones récurrents	44
	6 Choix des hyperparametres	45
	7 Formes de filtres	46
	8 Avantages de CNN	46
	9 Apprentissage par renforcement	47
	9.1 processus markovien	48
	9.2 Politique suivie par un agent	48
	9.2.1 Récompense reçue par un agent	48
	9.3 fonction de valeur et équation de Bellman	48
	9.4 Exploration versus exploitation	49
	9.5 principaux algorithmes d'AR	49
	9.5.1 choix d'algorithme	49
	9.5.1.1 Q-learning	49
	9.5.1. description	50
	9.6 Algorithme	50
	9.6.1 fonction mémoire	51
	9.6.2 fonction de renforcement	51
	9.6.3 paramètre d'apprentissage	51
	10 Conclusion	52
Chapitre 04	Expériences, Résultats et Discussions	53
	1 Introduction	54
	2 Environnement du travail	54
	3 Environnement matériel	54
	3.1 langage de programmation	54
	3.2 Tensorflow	55
	3.2.1 pourquoi Tensorflow ?	55
	3.3 Keras	56
	3.3.1 installation et compatibilité	56
	3.4 Dataset pour l'entrainement	56
	3.4.1 qu'est ce que Keggal ?	56
	3.5 ImageNet	57
	3.5.1 Historique	57
	3.6 ResNet50	58
	3.7 VGG	59
	3.7.1 VGG16 ?	59
	3.7.2 un peu de pratique	60
	3.8 Gradio	62
	3.8.1 installation	62
	4 descriptions de CNN	63
	5 L'architecture de l'application	64
	6 Base d'apprentissage	65
	7 l'apprentissage par CNN	65
	7.1 sélectionne des images IRM	66
	7.2 sauvegarde les paramètres de l'apprentissage	66
	7.3 la base de données	66
	7.4 résultat et discussion	67
	7.5 matrice de confusion	68
	Comment savoir si un modèle fait de l'overfetting	74
	Comment éviter l'overfetting	74
	Cross validation	74
	Matrice de confusion du modèle Resnet50	75
	Étude comparative entre les modèles (CNN, VGG16 et ResNet50)	76
	8 l'Apprentissage par renforcement	77
	8.1 résultats et discussion	77
	9 interface de l'application	82
	10 comparaisons de résultat avec d'autre travail	83
	11 conclusion	84
Conclusion générale		86
bibliographie		89

Liste des figures

Figure 1	évolution des décès dus au coronavirus janvier 2022	12
Figure 1.1	Une image montrant la forme du coronavirus	17
Figure 1.2	Symptôme communs	18
Figure 1.3	Taux de létalité en fonction d'âge	20
Figure 1.4	Taux de létalité en fonction d'âge	20
Figure 1.5	Montre le Nbre total cumulé de cas de covid-19 recensés en Algérie	24
Figure 1.6	Montre le Nbre de nouveaux cas quotidiens de covid19 en Algérie	25
Figure 1.7	Montre le Nbre total de décès au covid-19 en Algérie	25
Figure 1.8	Montre le Nbre de nouveau décès attribués covid-19 par jours en Algérie	25
Figure 2.1	Image noir et blanc	30
Figure 2.2	Une image en niveau de gris	31
Figure 2.3	Sous image de taille 5*5	31
Figure 2.4	Montre avec IRM l'état des poumons	33
Figure 2.5	IRM	33
Figure 2.6	Image IRM représentatives de la normale image	34
Figure 2.7	Image IRM représentatives de patient Covid-19	34
Figure 3.1	La relation entre IA,ML et le deep learning	35
Figure 3.2	Perception multicouche	39
Figure 3.3	Les réseaux de neurones convolutifs	40
Figure 3.4	Les couches de CNN	41
Figure 3.5	Principes de filtre convolutionnel	42
Figure 3.6	Principes du Pooling	43
Figure 3.7	Principe de la couche entretient connectées	43
Figure 3.8	L'étiquette codée de la couche de sortie CNN	44
Figure 3.9	Un RNN	45
Figure 3.10	Principe de l'apprentissage par renforcement	47
Figure 4.1	Logo pyton	55
Figure 4.2	Logo Tensorflow	55
Figure 4.3	Logo Keras	56
Figure 4.4	Extrait de l'architecture d'un réseau neurones convolutif	58
Figure 4.5	Architecture algorithme VGG16	59
Figure 4.6	Structure algorithme VGG16	59
Figure 4.7	Structure CNN	63
Figure 4.8	Architecture de l'application	64
Figure 4.9	Apprentissage image	65
Figure 4.10	Sélection un fichier IRM	66
Figure 4.11	Exemple image IRM	66
Figure 4.12	Résultat de détection	67
Figure 4.13	Modèle Accurcy et loss	68
Figure 4.14	Matrice de confusion	71
Figure 4.15	Montre le résultat du training du modèle vgg16	72
Figure 4.16	Les graphes du modèle vgg16	73
Figure 4.17	Résultat de la matrice de confusion vgg16	73
Figure 4.18	Cross validation	74
Figure 4.19	Matrice de confusion du modèle ResNet50	75
Figure 4.20	Illustration de l'overfetting	75
Figure 4.21	Graphe du modèle PPO	78
Figure 4.22	Graphe du modèle ppo montre nombre de temps	79
Figure 4.23	Graphe du modèle ppo montre la récompense	79
Figure 4.24	Graphe DQN	80
Figure 4.25	Graphe PPO	81
Figure 4.26	interface	82
Figure 4.27	Image IRM	82

LISTE DES ABREVIATIONS

AI	Intelligence Artificielle
CNN	Convolutional Neural Network
DL	Deep Learning
FN	False Negative
FP	False Positive
GUI	Graphical User Interface
ML	Machine Learning
NLP	Natural Language Processing
Pixels	Picture Element
RGB	Red Green Blue
RVB	Rouge Vert Bleu
TN	True Negative
TP	True Positive
TROD	test rapide d'orientation diagnostique
PCR	réaction en chaîne par polymérase
IRM	L'imagerie par résonance magnétique
TDM	Tomodensitométrie
EPFL	École polytechnique fédérale de Lausanne
ECDC	Centre européen de prévention et de contrôle des maladies, en anglais : <i>European Centre for Disease Prevention and Control</i>

Introduction

Générale



1-Introduction Générale

Maladie à virus Corona 2019 ou COVID-19 en abrégé, également connue sous le nom de maladie respiratoire aiguë associée au virus corona émergent. Ce virus est très proche du virus du SARS. Le virus a été découvert pour la première fois dans la ville chinoise de Wuhan en 2019. En raison de l'énorme économie de la Chine, cela a entraîné le déplacement de sa population vers tous les pays du monde. En conséquence, les peuples du monde ont été infectés et des millions de personnes ont été perdues à cause de ce virus. [1]

Avec l'accélération du nombre de cas dans le monde développe, les gouvernements prennent lentement conscience de l'ampleur du problème, et on pourrait espérer voir le monde s'unir contre une telle menace. Cependant, tant au niveau national qu'individuel, nous observons toujours des divisions et de la méfiance. Les rapports d'attaques racistes, de vols et de comportements irresponsables sont monnaie courante. Les médecins supplient le public de prendre la situation au sérieux et de s'abriter sur place tout en évitant tous les déplacements non essentiels.

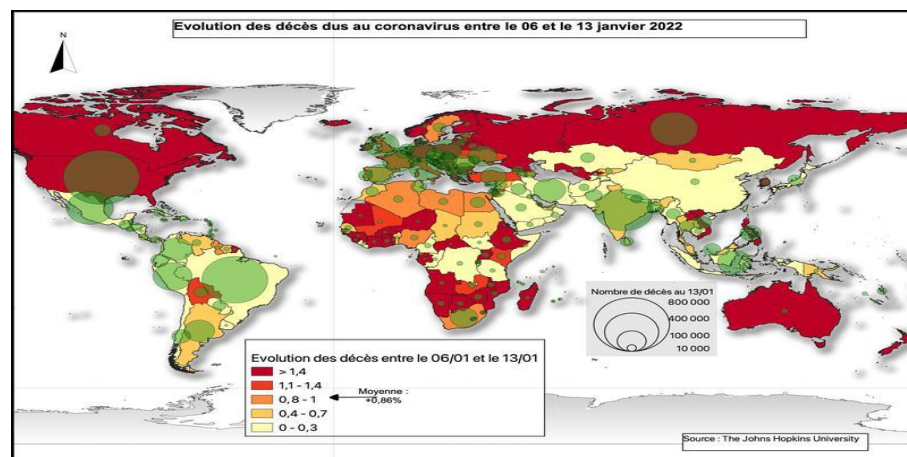


Figure 1: évolution des décès dus au coronavirus janvier 2022.

Les stratégies de lutte contre le virus avant mars ont principalement tourné autour du confinement et du traçage { pour trouver et isoler les cas possibles avant qu'ils ne se développent. Cependant, avec l'augmentation exponentielle des cas, les hôpitaux du monde entier sont rapidement devenus débordés et les politiques gouvernementales se sont orientées vers l'atténuation ou même l'acceptation. Le processus frustrant est encore aggravé par un manque de capacité de test dans les pays touchés, ce qui entraîne un retard du nombre de cas confirmés par rapport à la vraie valeur [1].

Cela a conduit les autorités sanitaires chinoises à revenir temporairement au diagnostic basé sur les symptômes et les radiographies/TDM thoraciques. Cependant, de telles solutions risquent toujours de se chevaucher avec d'autres causes de pneumonie, elles ont conduit à beaucoup de confusion et la nation est revenue à la PCR (réaction en chaîne par polymérase) - des tests basés sur la PCR [1].

Des solutions basées sur l'IA ont récemment été largement explorées pour l'utilisation de la détection de la pneumonie dans les études, et de telles approches ont également été proposées pendant cette pandémie [1].

Par conséquent, nous avons fait ce travail pour faciliter la détection de la maladie coronavirus en utilisant des images médicales, qui sont des Imagerie par résonance magnétique (IRM). Basé sur un sous-domaine de l'IA, les réseaux de neurones convolutifs (en anglais CNN ou ConvNet pour Convolutional Neural Networks), ils constituent les fers de lance de l'Apprentissage profond et l'apprentissage par renforcement (AR).

Aujourd'hui, les réseaux de neurones convolutifs et l'apprentissage par renforcement sont même capable d'apprendre à trier des images par catégorie avec dans certains cas, de meilleurs résultats qu'après un triage manuel. S'il y a donc aujourd'hui une méthode qui justifie un engouement particulier, il s'agit donc bien des CNNs. Ce qui est particulièrement intéressant avec les CNNs, c'est qu'ils sont également faciles à comprendre, lorsque vous les divisez en leurs fonctionnalités de base, c'est ce dont nous avons besoin dans notre projet. Nous voulons un moyen qui nous permette de faire la différence entre une personne malade du COVID -19 et une personne en bonne santé(NORMAL) sur la base de l'Imagerie par résonance magnétique (IRM).[1]

Afin d'atteindre ce but en a divisé le mémoire en quatre chapitres :

Chapitre 1: Coronavirus(COVID-19) - nous avons parlé du virus Corona, son origine, ses types et de sa propagation.

Chapitre 2: Représentation des Images - nous avons parlé du traitement d'images médicales à l'aide de la classification des images radiographiques thoraciques(IRM) et de la relation entre le virus Corona et les images.

Chapitre 3: nous présentons l'apprentissage en profondeur (Deep learning) et l'apprentissage par renforcement(AR) et en expliquons l'architecture de CNN.

Chapitre 4: nous présentons les résultats de ce travail. Nous avons comparé notre résultat avec les résultats d'autres travaux.

Ce mémoire se termine par une conclusion générale.

L'apprentissage profond s'appuie sur un réseau de neurones artificiels inspirés du cerveau humain. Ce réseau est constitué de dizaines et de centaines de "couches" de neurones, qui reçoivent et traduisent chacune les informations de la couche précédente. Par exemple, le système apprendra à reconnaître les caractères avant de traiter des mots dans du texte ou une image IRM , ou à déterminer s'il existe un visage sur une image avant que quiconque ne soit découvert.

L'objectif principal de notre travail est de détecter COVID-19 sur des images IRM utilisant les techniques de l'apprentissage profond et l'apprentissage par renforcement.

Coronavirus (COVID-19)

Chapitre 01



1 Introduction

En 2019, le monde entier est confronté à une urgence sanitaire en raison de l'émergence récente du coronavirus (COVID-19). À propos 196 pays sont touchés par le coronavirus, tandis que les États-Unis d'Amérique, l'Italie, la Chine, l'Espagne, l'Iran et la France ont le plus grand nombre de cas actifs de Covid19. Les cas et les services médicaux et de santé sont confrontés à des retards dans la détection du COVID-19. Plusieurs systèmes artificiels basés sur l'intelligence sont conçu pour détecter automatiquement le COVID-19 à l'aide des images médicales.[3]

2 Le coronavirus et le COVID-19?

2.1 Définition

Covid-19 fait référence à « *Coronavirus Disease 2019* », la maladie provoquée par un virus de la famille des *Coronaviridae*, le SARS-CoV-2. Cette maladie infectieuse est une zoonose, dont l'origine est encore débattue, qui a émergé en décembre 2019 dans la ville de Wuhan, dans la province du Hubei en Chine. Elle s'est rapidement propagée, d'abord dans toute la Chine, puis à l'étranger provoquant une épidémie mondiale.

Le Covid-19 est une maladie respiratoire pouvant être mortelle chez les patients fragilisés par l'âge ou une autre maladie chronique. Elle se transmet par contact rapproché avec des personnes infectées. La maladie pourrait aussi être transmise par des patients asymptomatiques mais les données scientifiques manquent pour en attester avec certitude[3]

2.2 Étiquette et forme :

Le nom «coronavirus» est dérivé de (latin: corona) et (grec: κορώνη) («koró nē», qui signifie couronne ou couronne), et signifie également couronne ou halo. Le nom fait référence à l'apparence caractéristique des virions (la forme infectieuse du virus) qui apparaît au microscope électronique, car ils ont un feuillage / villosités de grandes saillies de surface bulbeuses, les montrant comme une couronne de roi ou une couronne solaire(Figure1). Cette formation se produit via les fibrinogènes de l'épingle virale (S), qui sont des protéines qui remplissent la surface du virus et déterminent le tropisme d'un hôte.



Figure 1.1: Une image montrant la forme du coronavirus [4]

2.3 Les symptômes de la maladie Covid-19 :

Les symptômes du COVID-19 sont similaires à ceux d'autres rhumes et grippes et comprennent[5] :

2.3.1 Symptômes les plus fréquents

- fièvre
- toux
- fatigue
- perte de l'odorat ou du goût

2.3.2 Symptômes moins fréquents :

- maux de gorge
- maux de tête
- courbatures
- diarrhée
- éruption cutanée, ou décoloration des doigts ou des orteils
- yeux rouges ou irrités

2.3.3 Symptômes graves :

- difficultés à respirer ou essoufflement
- perte d'élocution ou de motricité, ou état confusionnel
- douleur au niveau de la poitrine.[5]

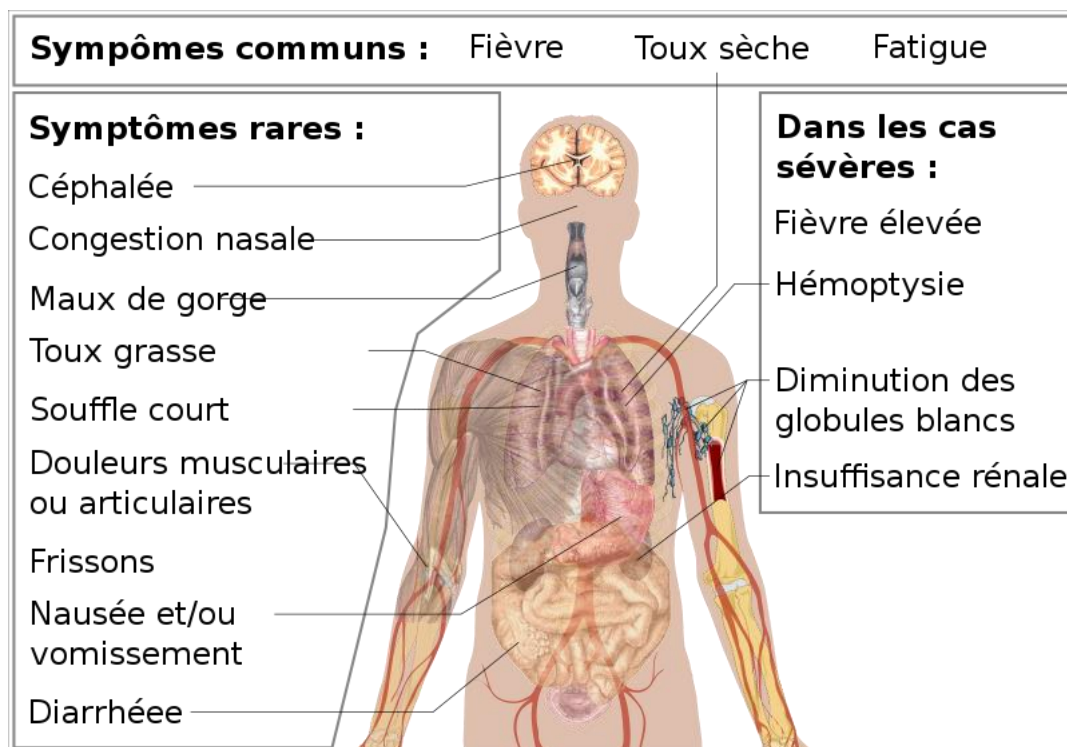


Figure:1. 2: **Symptômes communs**[6]

2.4 Les types de coronavirus : SRAS, MERS, Sars-CoV-2...

Il existe quatre sous-groupes principaux de coronavirus, appelés alpha, bêta, gamma et delta. Et sept formes différentes dont quatre communes (moins graves que les autres) :

- 229E (alpha coronavirus)
- NL63 (alpha coronavirus)
- OC43 (bêta-coronavirus)
- HKU1 (bêta-coronavirus)

Et trois plus graves :

- SRAS-CoV (le bêta-coronavirus qui cause le Syndrome respiratoire aigu sévère, ou SRAS, identifié en Chine en 2002).
- MERS-CoV (le coronavirus bêta qui provoque le syndrome respiratoire du Moyen-Orient, ou MERS découvert en 2012 en Arabie Saoudite).
- Sars-SoV-2 ou Covid-19 est pour la première fois identifié en Chine (Wuhan) en décembre 2019. Ce coronavirus a beaucoup de similitudes avec celui du SRAS (origine animale, identique génétiquement à 80%, responsable d'infections pulmonaires) mais aussi des différences notables pour les scientifiques au niveau de sa contagion. Il est contagieux dès le début des symptômes voire parfois en l'absence de symptômes alors que le Sras l'était quelques jours après les premiers symptômes. Il existe aussi des formes bénignes et asymptomatiques du Covid-19 alors que le Sras n'entraînaient que des formes sévères.[5]

2.5 Transmission

Les coronavirus se transmettent d'homme à homme lors de **contacts rapprochés** (se toucher ou se **serrer la main** par exemple) et par **voie aérienne** en **toussant** ou en **éternuant**. Toucher un objet ou une surface avec le virus dessus, puis toucher la bouche, le nez ou les yeux avant de se laver les mains peut aussi transmettre le coronavirus. Enfin, plus rarement la contamination peut se faire par contact fécal.

les coronavirus survivent jusqu'à 3 heures sur des surfaces inertes sèches et jusqu'à 6 jours en milieu humide. Ainsi la transmission manuportée à partir de l'environnement est possible, rappelle le Haut Conseil de la Santé publique dans un avis rendu le 5 mars 2020.[7]

2.6 Taux de létalité par âges

En décembre 2020, la maladie tue principalement les personnes âgées, 78 % des victimes de la Covid-19 ayant 75 ans et plus. Ce taux monte à 92 % pour les personnes de 65 ans et plus. L'âge moyen des décès liés à la Covid-19 est de 81 ans, et l'âge médian de 84 ans. Les moins de 50 ans ne représentent que 1,9 % des décès liés à la maladie.[8]

Distribution des décès en fonction de l'âge	
Âge	(en % de cas)
75 ans ou plus	78 %
65 - 74 ans	14 %
45 - 64 ans	7 %
Moins de 45 ans	1 %

Figure 1.4 : Taux de létalité en fonction de l'âge.[8]

2.7 Mortalité

En juillet 2021, l'ECDC indique que la maladie a tué 3 986 982 personnes dans le monde dont 740 809 dans l'union européenne et l'espace économique européen, 605 526 aux États-Unis, 524 417 au Brésil, 402 728 en Inde, 233 689 au Mexique, 193 389 au Pérou, 137 925 en Russie et 128 231 au Royaume-Uni entre le 31 décembre 2019 et la semaine 26 de l'année 2021.

Au 24 novembre 2021, le nombre de morts officiellement dues au Covid-19 est de 5,2 millions, mais sur la période prise en compte, l'excès de morts par rapport aux années précédentes est de 20,1 millions ; l'écart entre ces deux nombres est variable selon les pays : +20% dans l'Union Européenne et en Amérique du Nord, +700% en Asie, +800% en Afrique[9]

2.8 Test

2.8.1 Test de dépistage virologique

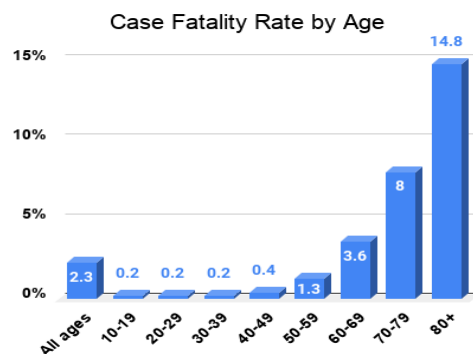


Figure 1.3 : Taux de létalité en fonction de l'âge.[8]

Les tests PCR(**de réaction en chaîne par polymérase**), ou tests de dépistage virologique, sont utilisés pour le diagnostic de l'infection COVID-19. Ils identifient la présence de gènes propres au SARS-CoV-2. Ils sont spécifiques et très fiables, si le prélèvement est bien réalisé. Les tests dits sérologiques sont, eux, destinés à identifier les personnes ayant développé une immunité contre le SARS-CoV-2.[10]

2.8.2 Tests sérologiques ?

Les tests sérologiques sont réalisés à partir d'un prélèvement sanguin. Ils recherchent des anticorps (protéines de l'immunité) témoins d'une infection passée. Ils ne sont donc pas adaptés au diagnostic de la COVID-19.

Certains de ces tests sont rapides et ne nécessitent que deux gouttes de sang (tests TROD).[10]

2.9 Examen d'imagerie thoracique ?

Les cliniciens pourraient utiliser l'imagerie thoracique pour diagnostiquer les personnes qui présentent des symptômes de la COVID-19, en attendant les résultats d'un test RT-PCR ou lorsque les résultats d'un test RT-PCR sont négatifs et que la personne présente des symptômes de la COVID-19.

Parmi les examens d'imagerie en trouve : Les rayons X , les scanners et IRM

produisent une image des organes et des structures du thorax.

- Les rayons X (radiographie) utilisent les radiations pour produire une image en 2-D. Généralement effectuées dans les hôpitaux, à l'aide d'un équipement fixe par un radiologue, elles peuvent également être effectuées sur des machines portables.
- La tomodensitométrie (TDM) utilise un ordinateur pour fusionner des images radiographiques en 2-D et les convertir en une image en 3-D. Elle nécessite un équipement hautement spécialisé et est réalisée à l'hôpital par un radiologue spécialisé.
- Les échographies utilisent des ondes acoustiques de haute fréquence pour produire une image. Elles peuvent être effectuées dans les hôpitaux ou dans d'autres établissements de soins, comme le cabinet d'un médecin.[11]

- Scanner: Dès le début de l'épidémie de COVID-19, le scanner thoracique sans injection s'est imposé comme l'examen d'imagerie pulmonaire de première intention en cas de diagnostic

suspecté ou confirmé de COVID-19, chez les patients présentant des signes de mauvaise tolérance respiratoire (dyspnée ou désaturation).

- IRM : Les Sociétés internationale et européenne de résonance magnétique en médecine (ISMRM et ESMRMB) tiennent leur congrès conjoint du 16 au 21 juin 2021 à Paris, « Le congrès montrera que l'IRM continue de progresser, que ce soit au niveau de la qualité d'image, de la rapidité des examens ou de la nature des informations collectées »[12].

3 Nouvelles technologies de diagnostic

3.1 Prise en charge et diagnostic par télé-médecine

La télé-médecine a montré son intérêt dans plusieurs situations de crises sanitaires. Elle pourrait permettre de trier les cas nécessitant une hospitalisation et pour les autres une surveillance à domicile. Le recueil des données associées avec un programme d'intelligence artificielle permet de trier et d'évaluer le risque. En cas d'hospitalisation, l'équipe médicale peut isoler et prendre en charge le patient de façon plus rapide, protégeant aussi les autres patients attendant dans le service des urgences[13]

3.2 Diagnostic par intelligence artificielle

Une étude scientifique semble montrer l'intérêt d'utiliser des logiciels d'IA, qui en analysant les anomalies pulmonaires visibles au scanner initial permettent de prédire avec de bonnes performances la durée d'hospitalisation des patients. Des chercheurs de l'EPFL ont développé une application web, CoughVid, ayant pour objectif de diagnostiquer la Covid-19 en analysant le son produit par la toux[14]

4 Traitement

En septembre 2021, il n'existe pas de traitement efficace ou de cure spécifique contre le SARS-CoV-2, reconnu par l'OMS. « Les soins de soutien les plus efficaces sont l'oxygénothérapie pour les patients dans un état grave et ceux qui risquent de développer une forme grave de la maladie, ainsi qu'une assistance respiratoire plus poussée, comme la ventilation pour les patients dans un état critique.

La dexaméthasone est un corticostéroïde qui peut contribuer à réduire la durée de placement sous respirateur et à sauver des patients dans un état grave ou critique »[15].

Le 4 novembre 2021, le **Royaume-Uni** devient le premier pays au monde à commercialiser le **molnupiravir**, un antiviral contre la COVID-19, pris sous forme de comprimés et destiné aux malades présentant un risque de forme grave. Ce traitement a été développé par le laboratoire américain **Merck**, commercialisé sous le nom de **Lagevrio**[15]

5 Vaccin

Le vaccin contre la maladie à coronavirus 2019 (Covid-19) fait l'objet de recherches dès le début de la pandémie de maladie à coronavirus 2019, dans plusieurs pays. Ainsi, le cofondateur de BioNTech a expliqué en novembre 2020, que son vaccin a été conçu en quelques heures dès janvier 2020

Différentes approches sont explorées pour forcer l'organisme à développer une réponse immunitaire permettant d'éliminer le virus SARS-CoV-2 : l'utilisation de formes atténuées ou inactivées du coronavirus, de virus anodins modifiés génétiquement pour exprimer une protéine du coronavirus, de protéines du coronavirus, ou d'acides nucléiques insérés dans des cellules humaines pour y produire des protéines virales.

En janvier 2021, selon l'Organisation mondiale de la santé (OMS), il y aurait soixante vaccins contre le coronavirus SARS-CoV-2 autorisés ou en phase d'étude clinique, ainsi que 172 vaccins potentiels à l'étude.

La Russie, la Chine et Cuba sont les premières à injecter à grande échelle un vaccin et à le commercialiser. Le vaccin russe Spoutnik V est le premier au monde à être homologué, son nom faisant référence au premier satellite envoyé dans l'espace. La Chine aurait déjà vacciné 350 000 personnes début octobre 2020.[16]

6 Pandémie de Covid-19 en Algérie

La pandémie de Covid-19 est une crise sanitaire majeure provoquée par une maladie infectieuse émergente apparue fin 2019 en Chine continentale, la maladie à coronavirus 2019, dont l'agent pathogène est le SARS-CoV-2. Ce virus est à l'origine d'une pandémie déclarée le 11 mars 2020 par l'Organisation mondiale de la santé. En Algérie, elle se propage à partir du 25 février 2020 lorsqu'un ressortissant italien est testé positif au SARS-CoV-2.[17]

À partir du 1er mars 2020, un foyer de contagion se forme dans la wilaya de Blida, seize membres d'une même famille ont été contaminés par le coronavirus lors d'une fête de mariage à

la suite de contacts avec des ressortissants algériens en France. Progressivement, l'épidémie se propage pour toucher toutes les wilayas algériennes.

D'autres cas de Covid-19 sont ensuite détectés. Dans un pays de 44 millions d'habitants il y a eu, au 7 août 2021 selon les chiffres officiels, un total de 180 350 malades, et un total de 4 510 décès.

6.1 Statistiques du début de la pandémie

Graphiques établis à partir des données recueillies depuis les communiqués de presse et les suivis quotidiens publiés par le ministère de la Santé, de la Population et de la Réforme hospitalière.

Au 20 mars 2020, l'Algérie détenait le taux de mortalité par le Coronavirus le plus élevé au monde avec près de 12 %.

Au 23 mars, 90 % des cas confirmés en Algérie ont été importés depuis l'Europe, principalement de la France et l'Espagne.

Au 26 mars, l'Algérie est le 64e pays le plus touché au monde en nombre de cas et 28e pays le plus touché au monde en nombre de décès-[17]

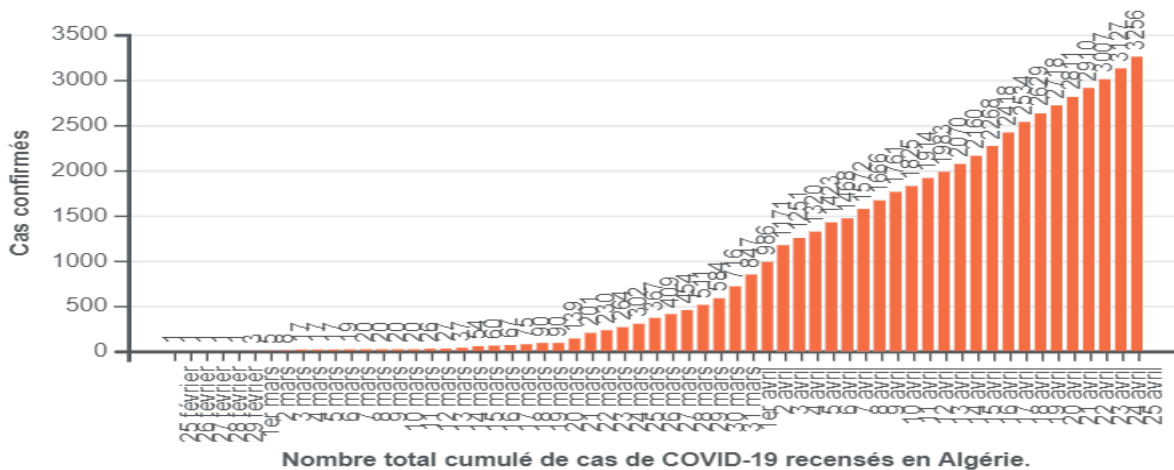


Figure : 1.5 montre le N^{bre} total cumulé de cas COVID-19 recensés en Algérie

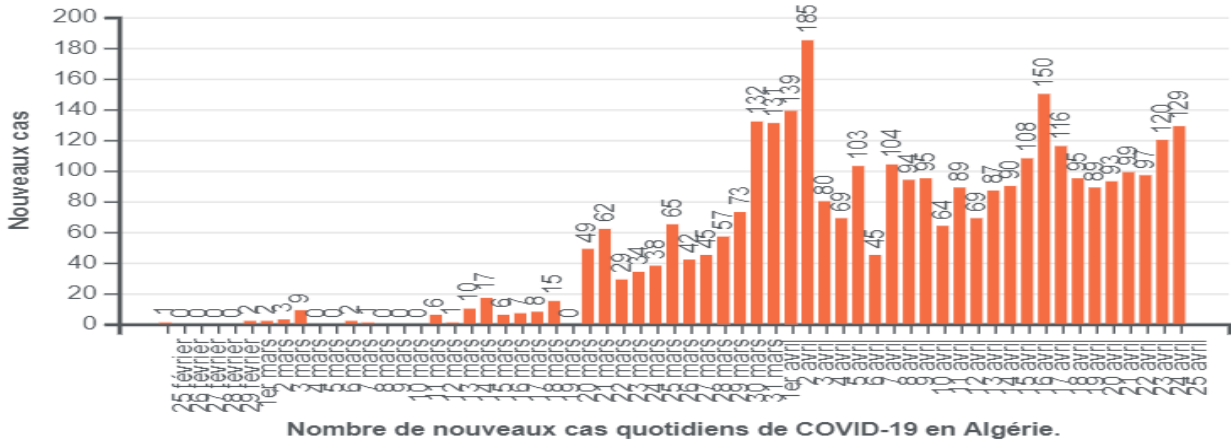


Figure1.6 : montre le N^{bre} de nouveaux cas quotidiens de COVID-19 en Algérie [17]

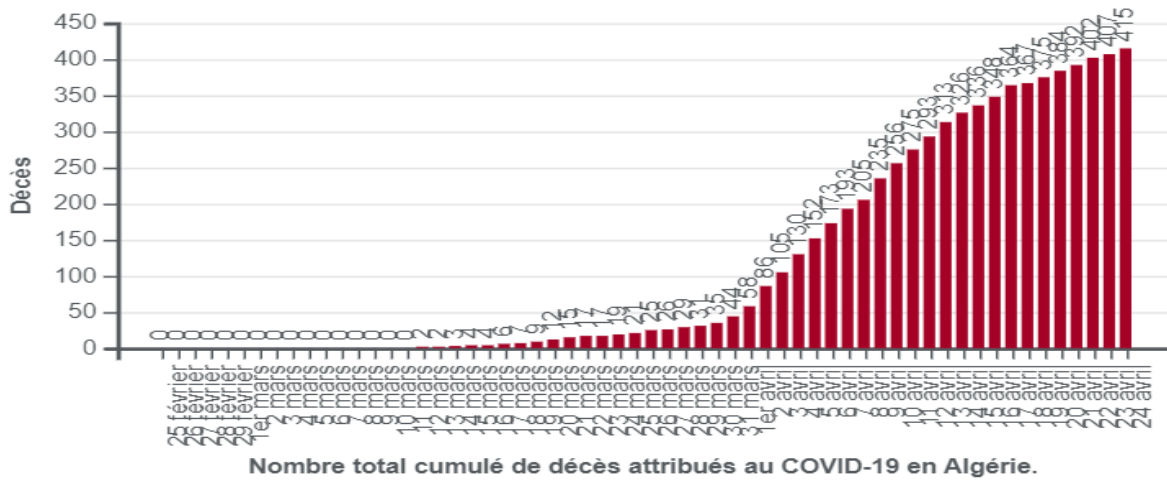


Figure1.7: montre le N^{bre} total de décès au COVID-19 en Algérie [17]

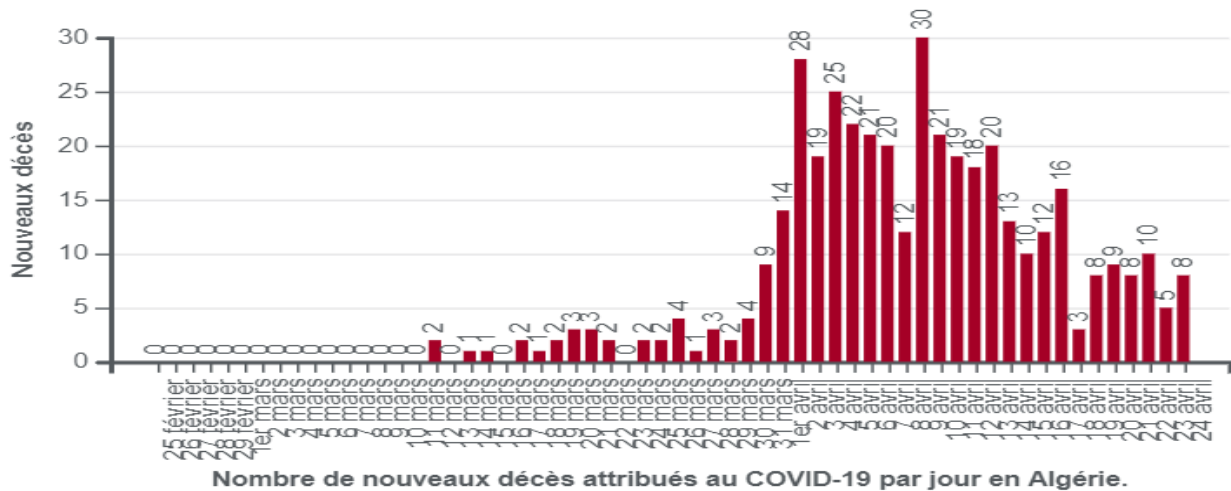
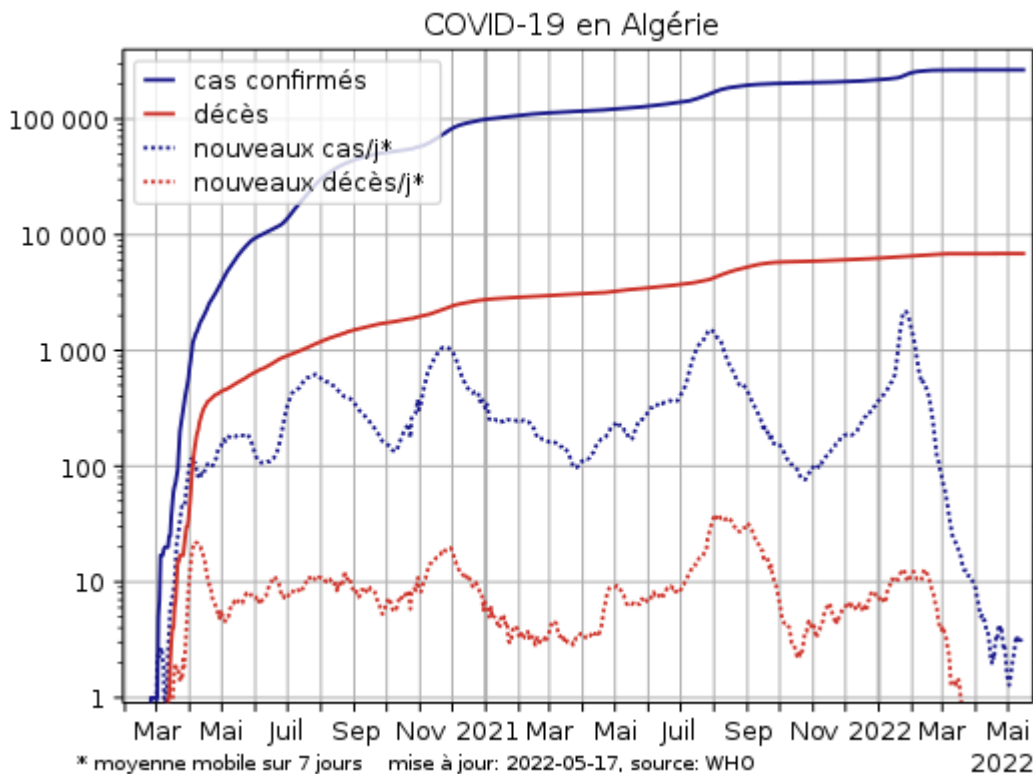


Figure1.8: montre le N^{bre} de nouveau décès attribués COVID-19 par jours en Algérie [17]

6.2 Historique

6.2.1 Chronologie



6.2.1.1-Première vague

Du 25 février 2020 au début Janvier 2021 l'Algérie franchit la barre des 100 000 cas confirmés. Au cours du mois 7 637 nouveaux cas et 137 décès, portent le nombre total de cas à 107 247 et celui de décès à 2 893.

En février ce sont 5 753 nouveaux cas et 92 décès, qui portent le nombre total de cas à 113 000 et donnent 2 985 décès.

6.2.1.2-Deuxième vague

Février 2021 au mai 2021 dépasse le nombre de cas à 129 013 et 3 472 décès.

6.2.1.3-Troisième vague

Du 03 mai au Avril 2022 Au cours du mois d'Avril il y a 111 nouveaux cas et un décès, portant le nombre de cas à 265 782 dont 6 875 décès[18]

Rapport de situation sur l'épidémie du covid-19 en ALGERIE " janvier 2022

Table 1 : Indicateurs épidémiologiques quotidiens en date du 07 janvier 2022 en Algérie

Item		Nombre
Nombre total de cas de COVID-19 confirmés en Algérie		221 316
Nombre de cas de COVID-19 confirmés dans les précédentes 24 heures		491
Nombre de wilayas ayant notifié pendant les dernières 24 heures :	0 nouveau cas de COVID-19	16
	Entre 1 et 9 nouveaux cas de COVID-19	17
	10 ou plus nouveaux cas de COVID-19	15
Nombre total de patients guéris du COVID-19 en Algérie		151 898
Nombre de patients guéris du COVID-19 dans les précédentes 24 heures		294
Nombre total de patients hospitalisés en soins intensifs pour COVID-19		31
Nombre total de décès par COVID-19 en Algérie		6 325
Nombre de décès par COVID-19 dans les précédentes 24 heures		7
Taux de létalité		2,86%
Taux de mortalité (sur 100 000 habitants)		9,26
Nombre de doses de vaccins COVID-19 reçues*		31 162 000
Nombre de doses de vaccins COVID-19 administrées*		12 760 747
Nombre de personnes vaccinées avec au moins 1 dose*		7 161 357
Nombre de personnes entièrement vaccinées*		5 699 263
Nombre de personnes vaccinées avec une dose de rappel*		99 012

*Données du Ministère de la Santé (situation arrêtée au 02 janvier 2022)[18]

7 Conclusion :

Le COVID-19 est difficile à combattre en raison de son comportement mystérieux et de sa biologie inconnue. Nous pouvons essayer des mesures de précaution et des leçons tirées d'autres flambées de santé publique comme le SRAS-CoV et Corona virus. la sensibilisation, la distanciation sociale, l'isolement, l'hygiène, la quarantaine et la vaccination peuvent réduire les opportunités de la propagation de l'épidémie, La vaccination doit toucher un public suffisamment large pour que l'immunité collective soit atteinte. Dans le cas contraire, le virus continue à circuler et à muter dans la population non vaccinée ce qui peut aboutir à l'apparition de variants de nature à compromettre l'efficacité des vaccins. Une étude montre que ce risque est maximal avec une couverture vaccinale de l'ordre de 60 % (situation de beaucoup de pays développés à l'été 2021).

Représentation des Images

Chapitre 02



1 L'Imagerie numérique

1.1 Définition de l'image numérique

Le terme d'image numérique désigne, dans son sens le plus général, toute image qui a été acquise, traitée et sauvegardée sous une forme codée représentable par des nombres (valeurs numériques).

La numérisation est le processus qui permet de passer de l'état d'image physique (image optique par exemple) qui est caractérisée par l'aspect continu du signal qu'elle représente (une infinité de valeur dans l'intensité lumineuse par exemple), à l'état d'image numérique qui est caractérisée par l'aspect discret (l'intensité lumineuse ne peut prendre que des valeurs quantifiées en un nombre fini de points distincts). C'est cette forme numérique qui permet une exploitation ultérieure par des outils logiciels sur ordinateur.[19]

1.2 Caractéristiques de l'imge numérique:

Une image numérique est caractérisée par sa **définition** et sa **résolution**.

- La **définition** correspond à la donnée hauteur \times largeur exprimée en pixels.
- La **résolution** désigne le nombre de pixels par unité de longueur de l'image analogique. Elle permet de mesurer la qualité de l'image obtenue après numérisation : plus la résolution est élevée, meilleure est la qualité de l'image.

1.3 Comment un ordinateur voit une image ?

L'ordinateur traite une image comme une matrice d'entiers de taille $h \times w$, contenant les niveaux de gris de ses pixels, chaque pixel est codé par un nombre binaire pour un niveau de gris, ou par trois nombres binaires qui correspond à une nuance de rouge , de vert et de bleu (code RVB en anglais (RGB)). [20]

- En distingue trois type d'images:
 - 1- Image noir et blanc
 - 2- Image niveau de gris
 - 3- Image couleur

1.3.1 Image noir et blanc

Chaque pixel d'une image en noir et blanc est représenté soit par un 0 (pixel blanc), soit par un 1 (pixel noir). Pour transmettre une image en noir et blanc, on l'encode puis on transmet la version encodée, que l'on décode à l'arrivée pour reconstituer l'image. [21]

Comme le montre la figure[2.1]

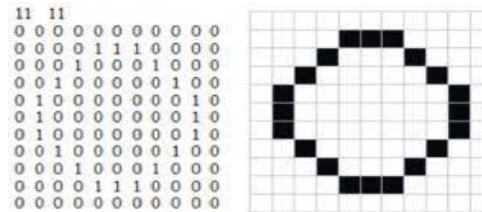


Figure 2.1 : Image noir et blanc

1.3.2 Image niveau de gris

Une image numérique en niveaux de gris est un tableau de valeurs. Chaque case de ce tableau, qui stocke une valeur, se nomme un pixel. En notant n le nombre de lignes et p le nombre de colonnes de l'image, on manipule ainsi un tableau de $n \times p$ pixels.

La figure ci-dessous montre une visualisation d'un tableau carré avec $n=p=240$, ce qui représente $240 \times 240 = 57600$ pixels.



Figure 2.2: Une image en niveaux de gris

Les valeurs des pixels sont enregistrées dans l'ordinateur sous forme de nombres entiers entre 0 et 255, ce qui fait 256 valeurs possibles pour chaque pixel.

La valeur 0 correspond au noir, et la valeur 255 correspond au blanc. Les valeurs intermédiaires correspondent à des niveaux de gris allant du noir au blanc.

La figure ci-dessous montre un sous-tableau de 5×5 pixels extrait de l'image précédente. On peut voir à la fois les valeurs qui composent le tableau et les niveaux de gris qui permettent d'afficher l'image à l'écran.[22]

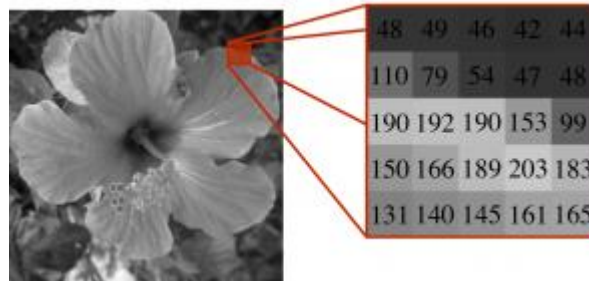


Figure 2.3: Sous image de taille 5×5

1.3.3 Image couleur

Une image numérique couleur est un fichier informatique pouvant être vu comme un tableau de nombres. Sa définition est le nombre total de points (pixels) qui la composent, et est donnée sous la forme nombre de pixels sur une ligne X nombre sur une colonne.

Le rendu des couleurs d'une image numérique peut être assuré par le codage RVB 24 bits. Pour chaque pixel de l'image, un triplet de nombres donne le niveau d'intensité lumineuse des trois sous-pixels rouge vert bleu formant le pixel. En 24 bits, cela permet d'obtenir plus de 16 millions de couleurs différentes.

Le codage RGB des couleurs est le codage informatique le plus proche de la réalité du mélange additif des couleurs.

Ce codage peut alors se faire sur 3 octets.

- ▶ RGB : Red-Green-Blue en anglais,
- ▶ RVB : Rouge-Vert-Bleu en français.

Chaque couleur est un mélange additif de ces trois couleurs souvent qualifiées de primaires. Sur un écran d'ordinateur, ce choix de couleurs primaires est issu de conventions technologiques définies par des normes.[23]

Le rouge est un nombre entier variant de 0 à 255

- ▶ Le vert est un nombre entier variant de 0 à 255
- ▶ Le bleu est un nombre entier variant de 0 à 255

Ainsi, une couleur va être repérée par un triplet (r,v,b) où chaque valeur est un nombre entier compris entre 0 et 255.

Mais pourquoi 255 ???

1.4 Le codage des couleurs sur 3 octets

Le codage d'un pixel peut se faire sur 32 bits, dont les 24 premiers bits sont utilisés pour coder la couleur.

Les 24 bits d'une couleur se décomposent donc en 3 fois 8 bits comme ceci :

- 8 bits sont consacrés à la teinte rouge ;
- 8 bits sont consacrés à la teinte vert ;
- 8 bits sont consacrés à la teinte bleu.

Le codage RGB de la couleur s'opère donc à l'aide de 3 octets de 8 bits.

Représentation d'un octet pour lequel tous les bits sont à 1 :

1	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---

255 est donc le plus grand nombre entier que l'on puisse coder sur 1 octet[23]

2 L'image médicale

2.1 Définition :

Faisant appel à de nombreuses technologies, l'imagerie médicale est de plus en plus utilisée pour le diagnostic de nombreuses maladies en complément d'un examen clinique et d'autres investigations, comme les examens biologiques, ou encore les tests neuropsychologiques.

L'imagerie médicale est aussi un élément essentiel à la recherche clinique, l'étude des maladies et la mise au point de nouveaux traitements. Il existe de nombreuses techniques d'imagerie complémentaires. L'imagerie recouvre à une grande variété de technologies développées grâce à l'exploitation des grandes découvertes de la physique du 20e siècle :

- Les ondes radio et rayons X
- La radioactivité de certains éléments
- Les champs magnétiques.[24]

2.2 les différentes technologies d'imagerie médicale

Radiographie | Scanner | Scintigraphie monophotonique | Imagerie par résonance magnétique (IRM). [24].

2.2.1 L'imagerie par résonance magnétique(IRM)

L'IRM ou Imagerie par résonance magnétique est l'une des techniques d'imagerie médicale les plus récentes. Elle permet de visualiser avec une grande précision les organes et tissus mous, dans différents plans de l'espace. Il est ainsi possible de déterminer la position exacte de lésions autrement invisibles. Réalisé sous la direction d'un médecin radiologue, cet examen ne provoque aucune irradiation. Il ne fait appel en effet, qu'aux propriétés des champs magnétiques.

Dans la mesure où l'IRM permet une analyse très fine de la structure des tissus, ses champs d'application sont larges. C'est ainsi que cet examen est particulièrement utilisé dans l'exploration des pathologies cérébrales, rachidiennes, osseuses, articulaires, digestives, gynécologiques, vasculaires et cardiaques.

2.2.2 Déroulement de l'IRM

L'appareil comporte un aimant de grande puissance, d'où le terme « magnétique ». La technique consiste à faire vibrer les noyaux d'hydrogène qui composent les tissus de l'organisme. Ces noyaux renferment en effet des protons qui vont agir comme autant de petits aimants. En vibrant, ils émettent des signaux qui vont être captés par une antenne puis être transformés en images. Dans la salle d'examen, le patient est allongé sur un lit. L'antenne est alors disposée sur la zone à examiner, puis le lit glisse doucement dans le tunnel de l'appareil. L'examen dure généralement de 15 à 20 minutes.[25]

La figure 2-4 montre avec IRM l'état des poumons d'un patient atteint du Covid 19 et de l'autre sain.



Figure 2-5: IRM

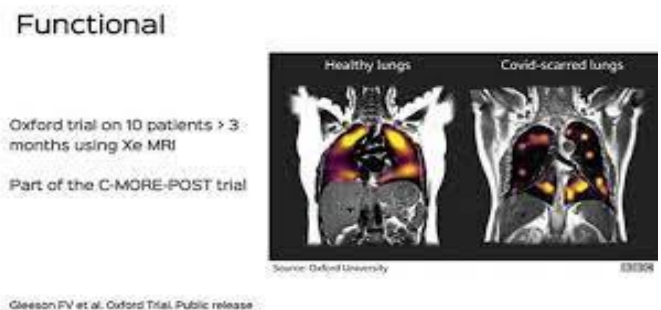


Figure 2.4 montre avec IRM l'état des poumons d'un patient atteint du Covid 19 et de l'autre sain.

3 La relation entre covid-19 et les images IRM :

Dans les techniques d'IA existantes, toutes les approches détectent la pneumonie causée par COVID-19 en utilisant des images de radiographie pulmonaire (database). [4] Parmi les défis auxquels l'IA est confrontée en ce qui concerne la détection de la pneumonie, il y a «comment le système réalise que la détection de la pneumonie dans la radiographie pulmonaire est causée par COVID-19 », de sorte que la pneumonie peut être causée par de nombreux autres virus, Bactérienne et par attaque fongique. Dans la pratique courante, tous les chercheurs s'accordent à poser le problème alors que «la majorité des décès dus de COVID-19 sont dus à une pneumonie dans les poumons de patients vulnérables », a déclaré le Dr Tom Naunton Morgan, médecin-chef de voici.ai. «La pneumonie est une maladie potentiellement mortelle causée par plusieurs agents pathogènes, y compris, directement ou indirectement, Infection au COVID-19 ». [26]

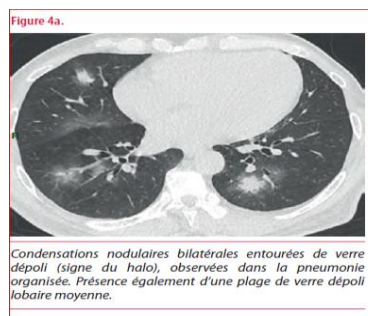


Figure 2.6: Images IRM représentatives de la normale Images

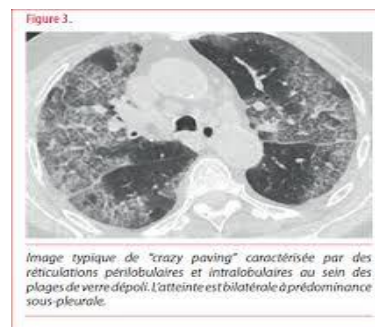
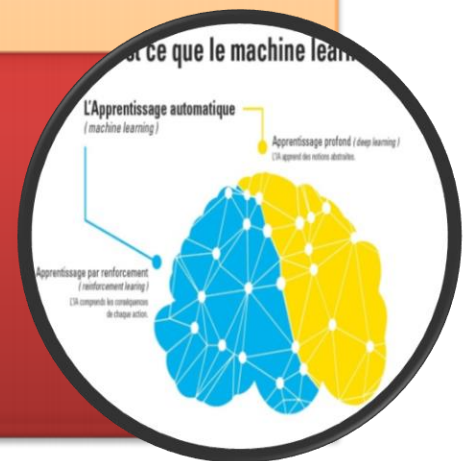


Figure 2.7 : Images IRM représentatives de patient COVID-19

L'apprentissage profond (Deep Learning) et apprentissage par renforcement

chapitre: 03



1 Introduction

L'apprentissage en profondeur est une branche de l'apprentissage automatique qui dépend entièrement de réseaux de neurones artificiels, car le réseau de neurones simulera le cerveau humain. L'apprentissage en profondeur est donc également un type de simulation du cerveau humain. En L'apprentissage profond, nous n'avons pas besoin de tout programmer explicitement. Le concept d'apprentissage en profondeur n'est pas nouveau. Il existe depuis quelques années maintenant. C'est un sujet de battage médiatique de nos jours car auparavant, nous n'avions pas beaucoup de puissance de traitement et beaucoup de données. Comme au cours des 20 dernières années, la puissance de traitement a considérablement augmenté, Dans ce chapitre nous allons présenter tout d'abord les notions en relation avec l'apprentissage profond.

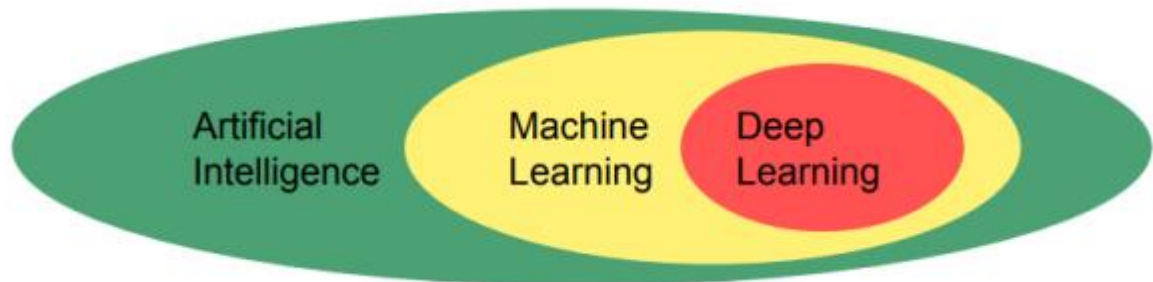


Figure 3.1: La relation entre l'intelligence artificielle, le ML et le deep learning[27].

2 Définition

L'apprentissage en profondeur est une fonction d'intelligence artificielle (IA) qui imite le fonctionnement du cerveau humain dans le traitement des données et la création de modèles à utiliser dans la prise de décision. L'apprentissage en profondeur est un sous-ensemble de l'apprentissage automatique dans l'intelligence artificielle qui dispose de réseaux capables d'apprendre sans surveillance à partir de données non structurées ou non étiquetées. Également connu sous le nom d'apprentissage neuronal profond ou de réseau neuronal profond [28].

3 Domaines d'application de l'apprentissage profonde

La technologie d'apprentissage en profondeur est l'une des techniques les plus utilisées dans de nombreux domaines, notamment:

3.1 La reconnaissance faciale

Il va s'agir en premier lieu de donner un certain nombre d'images à l'algorithme, puis à force d'entraînement, l'algorithme va être en mesure de détecter un visage sur une image.

3.2 Le traitement automatique de langage naturel

Le traitement automatique de langage naturel est une autre application du DL. Son but étant d'extraire le sens des mots, voire des phrases pour faire de l'analyse de sentiments.

3.3 Voitures autonomes

Les entreprises qui construisent de tels types de services d'aide à la conduite, doivent apprendre à un ordinateur à maîtriser certaines parties essentielles de la conduite à l'aide de systèmes de capteurs numériques au lieu de l'esprit humain. Pour ce faire, les entreprises commencent généralement par entraîner des algorithmes utilisant une grande quantité de données.

3.4 Recherche vocale et assistants à commande vocale

L'un des domaines d'utilisation les plus populaires de DL est la recherche vocale et les assistants intelligents à commande vocale. Avec les grands géants de la technologie ont déjà fait d'importants investissements dans ce domaine.

3.5 Traduction automatique

Il s'agit d'une tâche dans laquelle des mots, expressions ou phrases donnés dans une langue sont automatiquement traduits dans une autre langue. La traduction automatique existe depuis longtemps, mais DL permet d'obtenir les meilleurs résultats dans deux domaines spécifiques :

- Traduction automatique de texte
- Traduction automatique d'images

3.6 Recherche en marketing :

En plus de rechercher de nouvelles fonctionnalités susceptibles d'améliorer votre application, DL peut également être utile en arrière-plan. La segmentation du marché, l'analyse des campagnes marketing et bien d'autres peuvent être améliorés à l'aide de modèles de régression et de classification DL [27].

4 Présentation de quelques Types de réseaux neuronaux

Il existe beaucoup de types de réseaux neurones, chaque type étant développé pour un objectif particulier.[29]

4.1 Neurone Formel

Un neurone formel est une représentation mathématique et informatique d'un neurone biologique. Le neurone formel possède généralement plusieurs entrées et une sortie qui correspondent respectivement aux dendrites et au cône d'émergence du neurone biologique (point de départ de l'axone). Les actions excitatrices et inhibitrices des synapses sont représentées, la plupart du temps, par des coefficients numériques (les poids synaptiques) associés aux entrées. Les valeurs numériques de ces coefficients sont ajustées dans une phase d'apprentissage. Dans sa version la plus simple, un neurone formel calcule la somme pondérée des entrées reçues, puis applique à cette valeur une fonction d'activation, généralement non linéaire. La valeur finale obtenue est la sortie du neurone.

Le neurone formel est l'unité élémentaire des réseaux de neurones artificiels dans lesquels

il est associé à ses semblables pour calculer des fonctions arbitrairement complexes, utilisées pour diverses applications en intelligence artificielle.[29]

4-2 Neurones multicouche

Le perceptron multicouche (multi layer perceptron MLP) est un classifieur linéaire de type réseau neuronal formel organisé en plusieurs couches (Figure .1.4) au sein desquelles une information circule de la couche d'entrée vers la couche de sortie uniquement ; il s'agit donc d'un réseau de type feedforward. Chaque couche est constituée d'un nombre variable de neurones, les neurones de la couche de sortie correspondant toujours aux sorties du système [29].

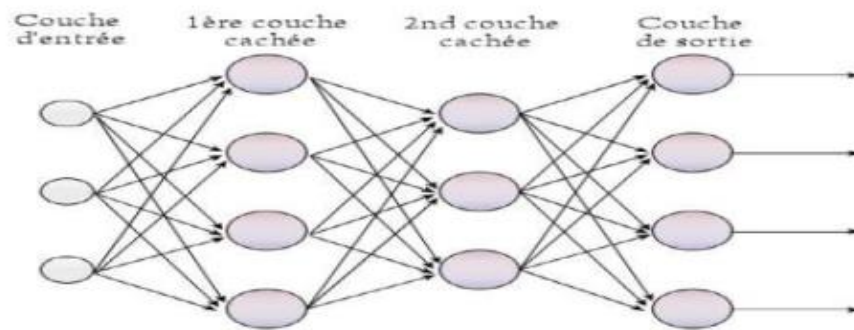


Figure 3.2: Perceptron multicouche[29]

4.3 Neurones récurrents

Les réseaux de Neurones récurrents (RNNs) permettent d'analyser les séquences de vecteurs tout comme les modèles de Markov cachés. Le temps entre ici en ligne de compte car les sorties (de la couche de sortie et/ou de la couche cachées) calculées à l'instant t sont ré-injectées en entrée du réseau et/ou en entrée de la couche cachée. On peut en théorie conserver dans le réseau la mémoire de ce qui s'y est passé depuis le début[29]

4.4 Réseaux de Hopfield

Le réseau de neurones d'Hopfield est un modèle de réseau de neurones récurrents à temps discret dont la matrice des connexions est symétrique et nulle sur la diagonale et où la dynamique est asynchrone (un seul neurone est mis à jour à chaque unité de temps). Il a été découvert par le physicien John Hopfield en 1982.

Sa découverte a permis de relancer l'intérêt dans les réseaux de neurones qui s'était essouffé durant les années 1970 à la suite d'un article de Marvin Minsky et Seymour Papert.

Un réseau de Hopfield est une mémoire adressable par son contenu : une forme mémorisée est retrouvée par une stabilisation du réseau, s'il a été stimulé par une partie adéquate de cette forme.[29]

4.5 Réseaux Neurones Convolutionnels

En apprentissage automatique, un réseau de neurone convolutifs (ou réseau de neurones à convolution, ou CNN ou ConvNet) est un type de réseau de neurones artificiels acycliques dans lequel le motif de connexion entre les neurones est inspiré par le cortex visuel des animaux. Les neurones de cette région du cerveau sont arrangés de sorte à ce qu'ils correspondent à des régions qui se chevauchent lors du pavage du champ visuel. Leur fonctionnement est inspiré par les processus biologiques, ils consistent en un empilage multicouche de perceptrons, dont le but est de prétraiter de petites quantités d'informations. Les réseaux neuronaux convolutifs ont de larges applications dans la reconnaissance d'image et vidéo, les systèmes de recommandation et le traitement du langage naturel.[29]

5 Architectures de réseaux de neurones profonds

Il existe un grand nombre de variables d'architectures profondes. La plupart d'entre eux sont dérivés de certaines architectures originales. Nous allons choisir les réseaux de neurones convolutifs (CNNs).

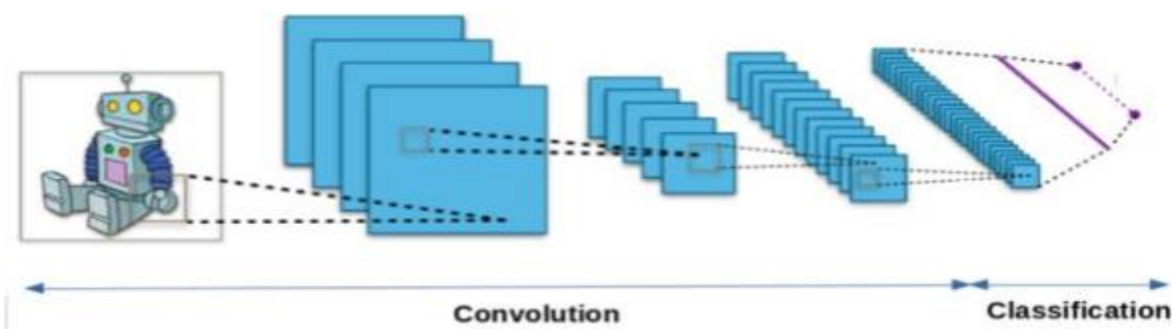
5.1 Les réseaux de neurones convolutifs

La vision par ordinateur évolue rapidement de jour en jour. Une des raisons est le développement de l'apprentissage en profondeur. Lorsque nous parlons de vision par ordinateur, un terme réseau de neurones convolutionnel. Désignés par l'acronyme CNN de l'anglais Convolutional Neural Network, ils comportent deux parties bien distinctes :

La première partie d'un CNN est la partie convolutive à proprement parler. Elle fonctionne comme un extracteur de caractéristiques des images. Une image est passée à travers d'une succession de filtres, ou noyaux de convolution, créant de nouvelles images appelées cartes de convolutions.

certains filtres intermédiaires réduisent la résolution de l'image par une opération de maximum local. En fin, les cartes de convolutions sont mises à plat et concaténées en un vecteur de caractéristiques, appelé code CNN.

Ce code CNN en sortie de la partie convolutive est ensuite branché en entrée d'une deuxième partie, constituée de couches entièrement connectées. Le rôle de cette partie est de combiner les caractéristiques du code CNN pour classer l'image. La sortie est une dernière couche comportant un neurone par catégorie. Les valeurs une distribution de probabilité sur les catégories [29].



5.1.1 Les couches de réseaux de neurones convolutionnels

Il y a plusieurs couches différentes dans CNN comme le montre la figure 3.4 :

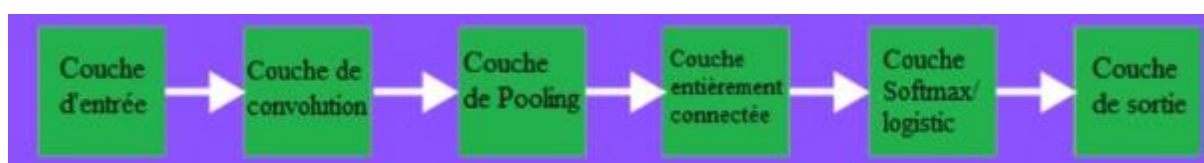


Figure 3.4 : Les couches de CNN [30]

5.1.2 La couche de convolution (CONV)

C'est la couche la plus importante et le cœur des éléments constitutifs du réseau convolutif, et c'est aussi elle qui effectue le plus de calculs lourds, car les caractéristiques de l'image sont extraites dans cette couche.

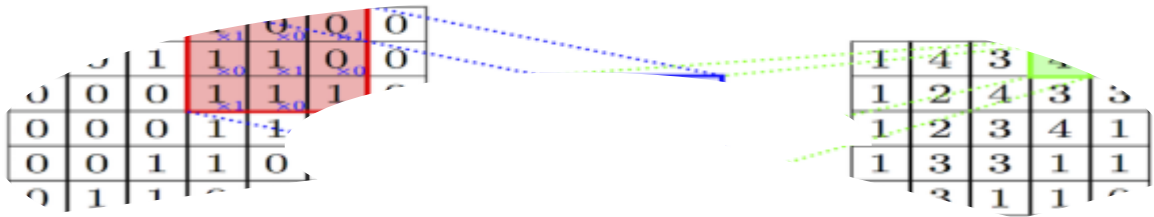


Figure 3.5: Exemple de principe du filtre convolutif[31]

5.1.3 La couche de Pooling

Un autre concept important des CNNs est le pooling (« mise en commun »), ce qu'est une forme de sous-échantillonnage de l'image. L'image d'entrée est découpée en une série de rectangles de n pixels de côté ne se chevauchant pas (pooling). Chaque rectangle peut être vu comme une tuile. Le signal en sortie de tuile est défini en fonction des valeurs prises par les différents pixels de la tuile. Le pooling réduit la taille spatiale d'une image intermédiaire, réduisant ainsi la quantité de paramètres et de calcul dans le réseau. Il est donc fréquent d'insérer périodiquement une couche de pooling entre deux couches convolutives successives d'une architecture CNN pour contrôler l'overfitting (sur-apprentissage). L'opération de pooling crée aussi une forme d'invariance par translation. La couche de pooling fonctionne indépendamment sur chaque tranche de profondeur de l'entrée et la redimensionne uniquement au niveau de la surface. La forme la plus courante est une couche de mise en commun avec des tuiles de taille 2x2 (largeur/hauteur) et comme valeur de sortie la valeur maximale en entrée (cf. schéma). On parle dans ce cas de « MaxPool 2x2 » (compression d'un facteur 4). Il est possible d'utiliser d'autres fonctions de pooling que le maximum. On peut utiliser un « average pooling » (la sortie est la moyenne des valeurs du patch d'entrée), du « L2-norm pooling ».

Dans les faits, même si initialement l'average pooling était souvent utilisé il s'est avéré que le max-pooling était plus efficace car celui-ci augmente plus significativement l'importance des activations fortes. En d'autres circonstances.[32]

Le pooling permet de gros gains en puissance de calcul. Cependant, en raison de la réduction agressive de la taille de la représentation (et donc de la perte d'information associée), la tendance actuelle est d'utiliser de petits filtres (type 2x2).

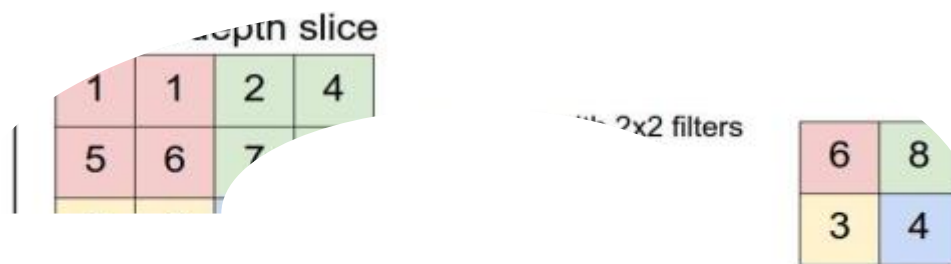


Figure 3.6 : Exemple de principe du Pooling

5.1.4 Couche entièrement connectée(FC)

Après plusieurs couches de convolution et de max-pooling le raisonnement de haut niveau dans le réseau neuronal se fait via des couches entièrement connectées [33]. Il connecte les neurones d'une couche aux neurones d'une autre couche. Il est utilisé pour classer les images entre différentes catégories par formation.



Figure 3.7:Principe de la couche entièrement connectée (FC) [34]

5.1.5 Couche Logistique ou Softmax :

Est la fin de la couche entièrement connectée .La logistique est utilisée pour la classification binaire et Softmax est pour la multi-classification.

5.1.6 Couche de sortie (output layer)

La couche de sortie contient l'étiquette qui est sous forme codée comme le montre la figure 3.8.

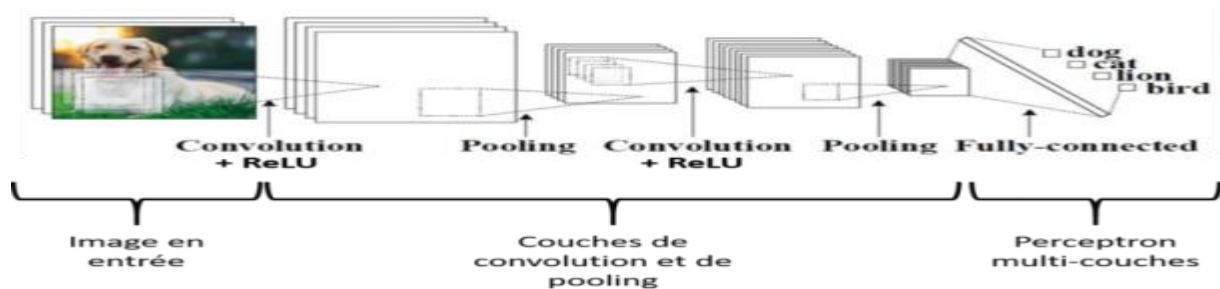


Figure 3.8 : Exemple montrant l'étiquette codée de la couche de sortie CNN

5.2 Réseau de neurones récurrents:

Les réseaux de neurones ne sont pas un modèle nouveau. En effet ils ont été introduits dès les années 50 – le perceptron est présenté en 1958, [33]. Et connaissent actuellement leur heure de gloire grâce aux architectures profondes – Deep learning – et à la puissance de calcul proposée par les GPU.

L'idée derrière les RNN est d'utiliser des informations séquentielles. Si on veut prédire le prochain mots qui dans une phrase, il faut connaître les mots sont venus avant . les RNN est qu'ils ont une «mémoire» qui capture des informations sur ce qui a été calculé jusqu'à présent.

En théorie, les RNNs peuvent utiliser les informations en séquence, Ils sont arbitrairement longs, mais en pratique ils se limitent à ne regarder que quelques pas en arrière. Voici à quoi ressemble un RNNs typique :

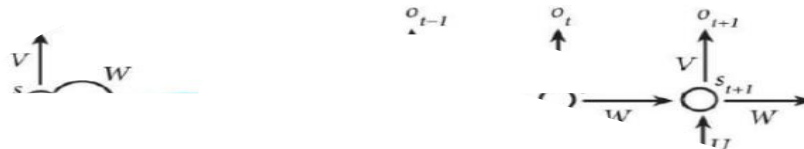


Figure 3.9: (à gauche) Un RNN. (à droite) Sa version déroulée [32]

Le schéma ci-dessus montre un RNN déroulé. En déroulant, nous signifions simplement qu'on montre le réseau pour la séquence complète. Par exemple, si la séquence qui nous intéresse est une phrase de 5 mots, le réseau serait déroulé en un réseau de neurones de 5 couches, une couche pour chaque mot. Les formules qui régissent les calculs dans un RNN sont les suivantes :

- X_t est l'entrée au moment t .
- $U; V; W$ sont les paramètres que le réseau va apprendre des données de l'apprentissage.
- S_t est l'état caché au moment t . C'est la « mémoire » du réseau. s_t est calculé en fonction de l'état caché précédent et de l'entrée à l'étape actuelle : [32]

$$S_t = f(Ux_t + Ws_{t-1})$$

6 Choix des hyperparamètres

Les CNNs utilisent plus d'hyperparamètres qu'un MLP standard. Même si les règles habituelles pour les taux d'apprentissage et des constantes de régularisation s'appliquent toujours, il faut prendre en considération les notions de nombre de filtres, leur forme et la forme du max pooling. II.4.4.1-Nombre de filtres : Comme la taille des images intermédiaires diminue avec la profondeur du traitement, les couches proches de l'entrée ont tendance à avoir moins de filtres tandis que les couches plus proches de la sortie peuvent en avoir davantage. Pour égaliser le calcul à chaque couche, le produit du nombre de caractéristiques et le nombre de pixels traités est généralement choisi pour être à peu près constant à travers les couches.

Pour préserver l'information en entrée, il faudrait maintenir le nombre de sorties intermédiaires (nombre d'images intermédiaire multiplié par le nombre de positions de pixel) pour être croissante (au sens large) d'une couche à l'autre. Le nombre d'images intermédiaires contrôle directement la puissance du système, dépend du nombre d'exemples disponibles et la complexité du traitement.[29]

7 Forme de filtres

Les formes de filtre varient grandement dans la littérature. Ils sont généralement choisis en fonction de l'ensemble de données. Les meilleurs résultats sur les images de MNIST (28x28) sont habituellement dans la gamme de 5x5 sur la première couche, tandis que les ensembles de données d'images naturelles (souvent avec des centaines de pixels dans chaque dimension) ont tendance à utiliser de plus grands filtres de première couche de 12x12, voire 15x15. Le défi est donc de trouver le bon niveau de granularité de manière à créer des abstractions à l'échelle appropriée et adaptée à chaque cas.[29]

8 Avantages de CNNs :

Outre sa fonction de filtrage, l'intérêt de la partie convolutive d'un CNN est qu'elle permet d'extraire des caractéristiques propres à chaque image en les compressant de façon à réduire leur taille initiale, via des méthodes de sous-échantillonnage tel que le Max-Pooling.

L'utilisation des CNN pour l'apprentissage en profondeur est devenue de plus en plus populaire en raison de trois facteurs importants.

- Les CNN éliminent le besoin d'extraction manuelle des fonctionnalités
- Les fonctionnalités sont apprises directement par le CNN .
- Les CNN produisent des résultats de reconnaissance de pointe.
- Les CNN peuvent être recyclés pour de nouvelles tâches de reconnaissance, ce qui vous permet de vous appuyer sur des réseaux préexistants .
- Les CNN offrent la possibilité de calculer automatiquement des cartes de caractéristiques (feature maps), évitant ainsi à l'utilisateur d'effectuer des calculs de caractéristiques lourds.[35]

9 Apprentissage par renforcement

L'apprentissage par renforcement, c'est apprendre à agir par essai et erreur. Dans ce paradigme, un agent peut percevoir son état et effectuer des actions. Après chaque action, une récompense numérique est donnée. Le but de l'agent est de maximiser la récompense totale qu'il reçoit au cours du temps.[36]

L'agent va donc recevoir des récompenses en fonction de « l'efficacité » des actions entreprises. Personne ne dit à l'agent quelle action il doit effectuer à un instant t . Grâce à ce système de récompense, l'agent devra découvrir par lui-même les actions à entreprendre en fonction de la situation. Nous avons donc, avec l'apprentissage par renforcement, un système basé sur la « recherche par essai-erreur ». Une action qui se révélera à plus ou moins long terme peu judicieuse sera sanctionnée par une récompense négative (par exemple -1). A contrario, si l'action effectuée s'avère positive (toujours à plus ou moins long terme), l'agent recevra une récompense positive (par exemple +1). À chaque pas de temps t , l'agent reçoit une représentation de l'état de l'environnement notée s_t . L'ensemble du processus est résumé par la Figure 3.[37]

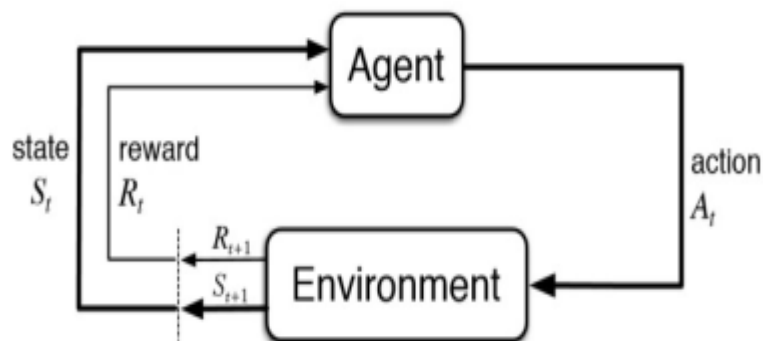


Figure 3.10: Principe de l'apprentissage par renforcement

Compte tenu de la dynamique des interactions avec l'environnement, le cadre mathématique adapté pour la description de l'AR dans le cas déterministe est donc le *modèle de processus de Markov*.

9.1 Processus markovien

On parlera d'état markovien si $P[s_{t+1}|s_t] = P[s_{t+1}|s_1, s_2, s_3, \dots, s_t]$. Autrement dit, la probabilité de se retrouver dans l'état s_{t+1} sachant que l'on se trouve dans l'état s_t , dépend uniquement de s_t et pas des états antérieurs (de s_1 à s_{t-1}). Dans un processus markovien, le futur dépend uniquement du présent et pas du passé ; remonter dans le passé n'apporte pas d'information. Dans le cadre de l'apprentissage par renforcement on utilisera des processus markovien.[36]

9.2 Politique suivie par un agent

L'agent, pour choisir l'action à effectuer depuis l'état s , va suivre une politique $\pi(s|a)$ qui pour un état s renvoie la probabilité de choisir l'action a : $\pi(s|a) = P(A = a|S = s)$ avec A l'ensemble des actions possibles et S l'ensemble des états possibles. L'agent va changer de politique au cours du temps en fonction de l'expérience acquise.[36]

9.2.1 Récompense reçue par un agent

On définit G_t la somme des récompenses futures qui seront reçues par l'agent depuis l'instant t . Nous avons :

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \dots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$$

$$V^\pi(s) = E[R_{t+1} + \gamma V^\pi(s_{t+1}) | s = s_t].$$

avec R les récompenses reçues aux différents instants et γ le taux d'escompte : une récompense reçue dans k pas de temps dans le futur vaut γ^k fois ce qu'elle aurait valu si elle était reçue immédiatement. Nous avons $0 \leq \gamma \leq 1$, plus γ est petit et moins la récompense reçue dans le futur aura de l'importance dans la politique choisie par l'agent.[36]

9.3 Fonction de valeur et équation de Bellman

$V^\pi(s)$ est la fonction de valeur, nous avons $V^\pi(s) = E[G_t | s = s_t]$, la fonction de valeur représente donc les récompenses espérées dans le futur, sachant que nous sommes actuellement dans l'état s_t . Nous avons donc aussi $V^\pi(s) = E[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} | s = s_t]$ soit après quelques lignes de calculs

On nomme cette équation l'équation de Bellman.

9.4 « Exploration versus Exploitation »

Le dilemme « Exploration versus Exploitation » est un aspect fondamental de l'apprentissage par renforcement. Faut-il choisir d'exécuter une série d'actions déjà explorée par l'agent au préalable où la récompense finale est déjà plus ou moins connue et ainsi exploiter les connaissances déjà acquises. Ou choisir d'effectuer des enchaînements d'action inconnus jusqu'alors avec, au bout, une récompense qui pourrait potentiellement être intéressante ?

9.5 Principaux algorithmes d'AR

Plusieurs algorithmes d'AR (SARSA, QP learning, AHC-learning, R-learning, critique-acteur, $TD(\lambda)$, $SARSA(\lambda)$, Q -learning(λ)...) se basent sur une comparaison entre récompense que l'on s'attend à recevoir et la récompense que l'on reçoit

Ces approches se distinguent par le type de fonction valeur estimée, et par les techniques d'évaluation et d'amélioration des stratégies.[36]

9.5.1 Choix de l'algorithme

notre choix s'est orienté vers le **Q-learning**, cet algorithme présente plusieurs atouts. Il a été introduit par Christopher Watkins en 1989 est sans doute l'algorithme d'apprentissage le plus utilisé pour plusieurs raisons : D'une part, cet algorithme est très simple et très ouvert d'autre part, sa convergence a été prouvée dans le cas des processus de décision markoviens .

Aussi, on effectue les mises à jour de la fonction de renforcement en fonction des actions optimales ce qui rend l'algorithme plus efficace. La solution optimale est atteinte plus rapidement par rapport aux algorithmes.

9.5.1.1 Q-learning

En intelligence artificielle, plus précisément en apprentissage automatique, le **Q-learning** est une technique d'apprentissage par renforcement. Cette technique ne nécessite aucun modèle initial de l'environnement. La lettre 'Q' désigne la fonction qui mesure la qualité d'une action exécutée dans un état donné du système[38]. C'est un algorithme off-policy.

Dans le Q-learning, l'agent exécute une action **a** en fonction de l'état **s** et d'une fonction **Q**. Il perçoit alors le nouvel état **s'** et une récompense **r** de l'environnement.

Il met alors à jour la fonction **Q**. Le nouvel état **s'** devient alors l'état **s**, et l'apprentissage continue.

9.5.1.2 Description

Cette méthode d'apprentissage permet d'apprendre une stratégie, qui indique quelle action effectuer dans chaque état du système. Elle fonctionne par l'apprentissage d'une fonction de valeur état-action notée Q qui permet de déterminer le gain potentiel, c'est-à-dire la récompense sur le long terme, $Q(s,a)$, apportée par le choix d'une certaine action a dans un certain état s en suivant une politique optimale. Lorsque cette fonction de valeur d'action-état est connue ou apprise par l'agent, la stratégie optimale peut être construite en sélectionnant

l'action à valeur maximale pour chaque état, c'est-à-dire en sélectionnant l'action a qui maximise la valeur $Q(s,a)$ quand l'agent se trouve dans l'état s . [39]

Un des points forts du Q-learning est qu'il permet de comparer les récompenses probables de prendre les actions accessibles sans avoir de connaissance initiale de l'environnement. En d'autres termes, bien que le système soit modélisé par un processus de décision markovien (fini), l'agent qui apprend ne le connaît pas et l'algorithme du Q-learning ne l'utilise pas.

Cette notion d'apprentissage par récompense a été introduite à l'origine dans la thèse de Watkins en 1989. C'est une variante de l'apprentissage par différence temporelle. Le Q-learning converge vers une stratégie optimale, c'est-à-dire qu'il conduit à maximiser la récompense totale des étapes successives. [40]

9.6 Algorithme

9.6.1 - Fonction mémoire :

A l'origine, la mémoire utilisait implicitement un tableau (statique ou dynamique) de dimension : nombre d'état (S) X nombre d'action (A) pour stocker la fonction d'utilité.

L'emploi de cette structure est requis pour prouver la convergence du Q-Learning.

Malgré tout, il est possible d'utiliser à la place d'un tableau une fonction d'approximation telle que fonctions polynomiales, statistiques et réseaux de neurones. [40]

9.6.2 -Fonction de renforcement :

Cette fonction est fondée sur l'algorithme d'itérations sur les valeurs dont l'objectif est d'optimiser la fonction d'utilité pour chaque couple état-action (s,a) afin d'en déduire une stratégie de commande optimale. Le Q-learning met à jour la fonction d'utilité .

9.6.3 Paramètre d'apprentissage :

Le facteur α ($0 < \alpha \leq 1$) est appelé le taux d'apprentissage et détermine l'importance de la correction réalisée sur la fonction Q lors d'une mise à jour. Deux conditions sur α garantissent à l'algorithme du Q-learning de converger vers la fonction de valeur optimale : $\sum_{t=0}^{\infty} \alpha^t < \infty$ [40]

10 Conclusion :

Dans ce chapitre on a présenté les notions importantes qui sont en relation avec l'apprentissage profond (définition, Architectures...etc). Aussi qu'une vision générale sur l'apprentissage profond, toute on donnant en détail la méthode choisie dans notre travail de recherche qui est le CNNs, nous avons aussi parlé de l'apprentissage par renforcement et ses techniques. Le prochain chapitre, traite les détails de la conception, ainsi que la méthode et les outils utilisés pour la réalisation de notre application.

Expériences, Résultats et Discussions

Chapitre 04



1 Introduction

Après avoir présenté la théorie de l'apprentissage profond et l'apprentissage par renforcement dans le chapitre précédent. Ce partie sera consacrée à la mise en œuvre d'un système pour la détection de Covid-19. Pour cela, nous avons utilisé un groupe d'images sélectionnées à partir de la base de données qui contient une collection d'images IRM. Nous avons développé deux modèles de transfert d'apprentissage à partir des réseaux CNN, VGG16 et ResNet50. Nous avons utilisé des modèles avec des techniques simples et que nous avons optimisés pour l'apprentissage.

2 Environnement du travail :

Dans cette section, nous présenterons le matériel et le logiciel utilisés dans notre travail.

3 Environnement matériel :

Afin de mettre en œuvre ce projet, nous avons utilisé un ensemble de matériel dont les caractéristiques sont les suivantes :

Un ordinateur hp avec les caractéristiques suivantes :

Processeur : Intel® *Core™* i5-3320 U CPU @2.60GHz 2.60 GHz

Mémoire installée (RAM) : 8Go

Type de système : système d'exploitation 64bits.

OS : Microsoft Windows 8 .

3.1 Langage de programmation(Python) :

Python a été développé à l'Institut néerlandais de mathématiques et d'informatique (CWI) à Amsterdam par Guido van Rossum à la fin des années 1980, et sa première annonce remonte à 1991. Le noyau du langage est écrit en langage C. Ross a appelé son langage "Python" pour exprimer son admiration pour un célèbre groupe de sketches britannique qui s'appelait Monty Python.[42]

Python est un langage de programmation de haut niveau conçu pour être facile à lire et à implémenter. Il est open source, ce qui signifie qu'il est gratuit, même pour des applications commerciales. Python peut fonctionner sur les systèmes Mac, Windows et Unix. [42]



Figure 4.1: Logo Python

3.2 TensorFlow :

TensorFlow est une plate-forme open source de bout en bout pour la création d'applications d'apprentissage automatique. Il s'agit d'une bibliothèque mathématique symbolique qui utilise un flux de données et une programmation différentiable pour effectuer diverses tâches axées sur la formation et l'inférence de réseaux de neurones profonds. Il permet aux développeurs de créer des applications d'apprentissage automatique à l'aide de divers outils, bibliothèques et ressources communautaires.[43]

Actuellement, la bibliothèque d'apprentissage en profondeur la plus célèbre au monde est TensorFlow de Google. Google utilise l'apprentissage automatique dans tous ses produits pour l'optimisation des moteurs de recherche, la traduction, les légendes d'images ou les recommandations



Figure 4.2: logo TensorFlow

3.2.1 Pourquoi TensorFlow ?

Que vous soyez expert ou débutant, TensorFlow est une plate-forme de bout en bout qui facilite la création et le déploiement de modèles ML. TensorFlow offre plusieurs niveaux d'abstraction afin que nous pouvons choisir celui qui convient le mieux à nos besoins.

Créez et entraînez des modèles à l'aide de l'API Keras de haut niveau, ce qui facilite la mise en route de TensorFlow et l'apprentissage automatique. [43]

Si vous avez besoin de plus de flexibilité, une exécution rapide permet une itération immédiate et un débogage intuitif. Pour les tâches de formation de grande taille, utilisez l'API de stratégie de distribution pour une formation distribuée sur différentes configurations matérielles sans modifier la définition du modèle.

3.3 Keras

L'une des bibliothèques Python les plus puissantes et les plus faciles à utiliser pour développer et évaluer des modèles d'apprentissage en profondeur est Keras ; Il englobe les bibliothèques de calcul numérique efficaces Theano et TensorFlow. L'avantage de ceci est principalement que vous pouvez vous lancer avec les réseaux de neurones de manière simple et amusante [44].



Figure 4.3: logo Keras

3.3.1 Installation et compatibilité

Keras est livré avec TensorFlow 2 en tant que `tensorflow.keras`. Pour commencer à utiliser Keras, **installez** simplement TensorFlow 2 [44].

Keras / TensorFlow sont compatibles avec :

- | Python 3.5{3.8
- | Windows 7 ou version ultérieure
- | macOS 10.12.6 (Sierra) ou version ultérieure
- | Ubuntu 16.04 ou version ultérieure.

3.4 Dataset pour l'entraînement

. Dataset pour l'entraînement

Nous avons téléchargé les dataset pour réaliser ce projet à partir plateforme Kaggel.

3.4.1 Qu'est-ce que Kaggel ?

Kaggle est une plateforme web organisant des compétitions en science des données. Sur cette plateforme, les entreprises proposent des problèmes en science des données et offrent un prix aux datalogistes obtenant les meilleures performances. L'entreprise a été fondée en 2010 par Anthony Goldbloom.[44]

3.5 ImageNet

Est une base de données d'images annotées produit par l'organisation du même nom, à destination des travaux de recherche en vision par ordinateur. En 2016, plus de dix millions d'URLs ont été annotées à la main pour indiquer quels objets sont représentés dans l'image ; plus d'un million d'images bénéficient en plus de boîtes englobantes autour des objets. La base de données d'annotations sur des URL d'images tierces est disponible librement, ImageNet ne possédant cependant pas les images elles-mêmes. De 2010 à 2017, le projet ImageNet a organisé un concours annuel : ImageNet Large Scale Visual Recognition Challenge (ILSVRC), ou "Compétition ImageNet de Reconnaissance Visuelle à Grande Échelle". Elle consistait en une compétition logicielle dont le but était de détecter et classifier précisément des objets et des scènes dans les images naturelles.[47]

3.5.1 Historique

La chercheuse Fei-Fei Li a commencé à travailler sur l'idée du dataset ImageNet en 2006. À une époque où les chercheurs se focalisent sur les modèles et les algorithmes, Li décide d'augmenter et d'améliorer les données disponibles pour entraîner les algorithmes d'apprentissage artificiel.[46] En 2007, Fei-Fei Li rencontre Christiane Fellbaum, professeure à l'Université de Princeton et une des créatrices de WordNet pour discuter du projet. À la suite de cette rencontre, Li construit ImageNet à partir de WordNet, en ré-utilisant plusieurs de ses caractéristiques.

La base de données ImageNet a été présentée pour la première fois lors des sessions de présentation de posters en 2009 à la Conférence sur la Vision par Ordinateur et Reconnaissance de formes (CVPR) en Floride, par des chercheurs du département d'Informatique à l'Université de Princeton.[47]

3.6 ResNet50

Ce modèle est composé de 50 couches, et a la particularité d'introduire des **connexions résiduelles**.

Contrairement aux réseaux de neurones convolutifs qui ont une architecture linéaire (un empilement de couches dont chaque sortie est uniquement connectée à la couche suivante [voir l'architecture A de la figure suivante]), dans un réseau résiduel, la sortie des couches précédentes est reliée à la sortie de nouvelles couches pour les transmettre toutes les deux à la couche suivante. Un schéma s'impose [voir l'architecture B de la figure suivante]:[48]

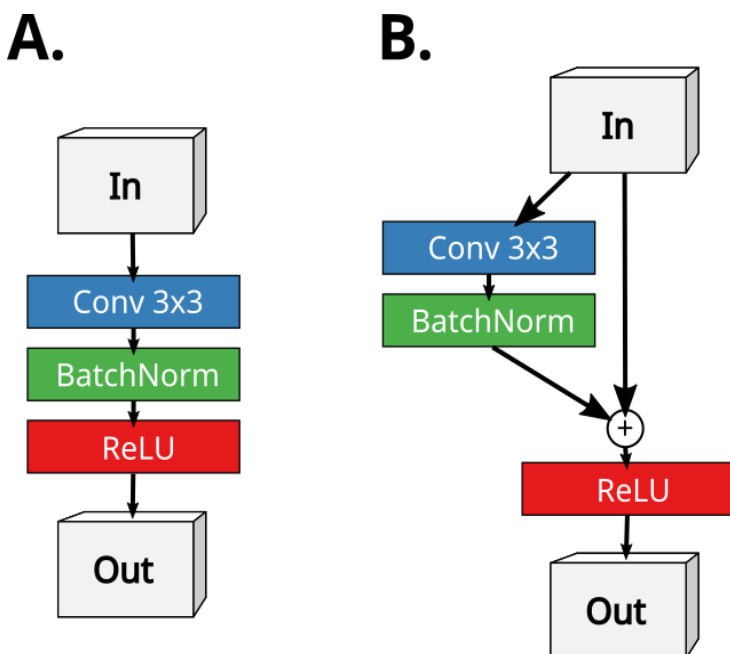


Figure 4.4: Extrait de l'architecture d'un réseau de neurones convolutif sans (A) et avec (B) connexion résiduelle

Avant la mise au point de ce type de réseau, il était impossible d'entraîner un réseau comportant plus de 25 couches. Les couches devenant plus profondes, les gradients devenant plus petits, les performances en étaient inévitablement dégradées : l'erreur ne se propageait plus correctement et la mise à jour des pondérations en était directement impactée.

Les réseaux de neurones résiduels ont permis d'aller au-delà de cette limitation. Leur architecture permet la création de **réseaux de neurones très profonds**, à la **précision meilleure** que ceux ayant des architectures linéaires car ils ont la capacité d'extraire davantage d'informations et d'avoir ainsi une analyse plus avancée des images.

Le modèle Resnet50 offre aussi l'avantage d'être directement **disponible dans Keras** (un de nos frameworks de prédilection avec scikit-learn), et de fonctionner plutôt bien pour la tâche que nous essayons de faire aujourd'hui.[48]

3.7 VGG

est un réseau de neurones convolutionnels proposés par K. Simonyan et A. Zisserman de l'université d'Oxford et qui a acquis une notoriété en gagnant la compétition **ILSVRC** (ImageNet Large Scale Visual Recognition Challenge) en 2014. Le modèle a atteint une précision de 92.7% sur Imagenet ce qui est un des meilleurs scores obtenus. Il a marqué une progression par rapport aux modèles précédents en proposant, dans les couches de convolution, **des noyaux de convolution de plus petites dimensions (3×3)** que ce qui avait été fait jusque-là. Le modèle a été entraîné sur des semaines en utilisant des cartes graphiques de pointe.[41]

3.7.1 VGG-16 ?

VGG16 est un modèle de réseau de neurones à convolution conçu par K. Simonyan et A. Zisserman. Ce modèle atteint une précision de test de **92,7%** dans ImageNet, qui regroupe plus de 14 millions d'images appartenant à 1000 classes. Pourquoi vgg-16 et bien tout simplement parce que ce réseau de neurones comprend 16 couches profondes :[41]

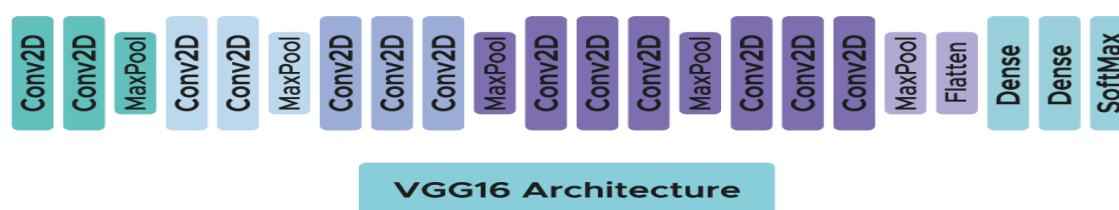


Figure 4.5: Architecture Algorithm VGG16

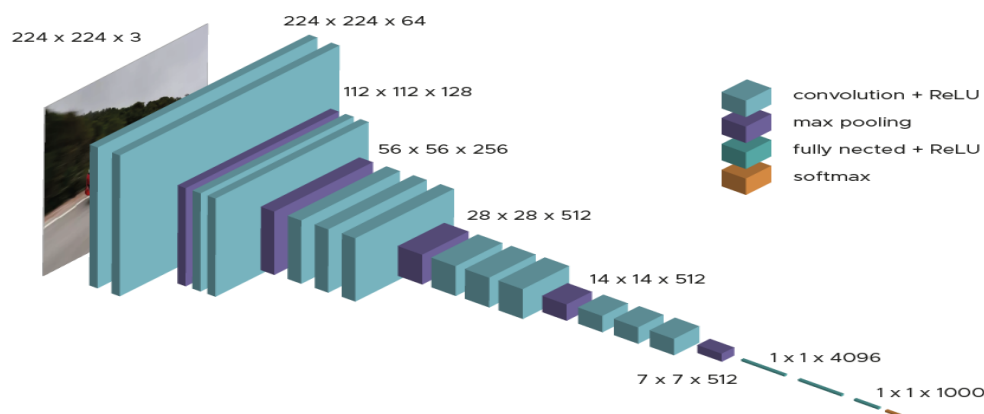


Figure 4.6: Structure Algorithme VGG16

Le modèle ne demande qu'un prétraitement spécifique qui consiste à soustraire la valeur RGB moyenne, calculée sur l'ensemble d'apprentissage, de chaque pixel.

Durant l'apprentissage du modèle, l'input de la première couche de convolution est une image RGB de taille 224 x 224. Pour toutes les couches de convolution, le noyau de convolution est de taille 3x3: la plus petite dimension pour capturer les notions de haut, bas, gauche/droite et centre. C'était une spécificité du modèle au moment de sa publication. Jusqu'à VGG16 beaucoup de modèles s'orientaient vers des noyaux de convolution de plus grande dimension (de taille 11 ou bien de taille 5 par exemple). Ces couches ont pour but de filtrer l'image en ne gardant que des informations discriminantes comme des formes géométriques atypiques. Ces couches de convolution s'accompagnent de couche de Max-Pooling, chacune de taille 2x2, pour réduire la taille des filtres au cours de l'apprentissage.

En sortie des couches de convolution et pooling, nous avons 3 couches de neurones Fully-Connected. Les deux premières sont composées de 4096 neurones et la dernière de 1000 neurones avec une fonction d'activation softmax pour déterminer la classe de l'image.

Comme vous avez pu le constater l'architecture est claire et simple à comprendre ce qui est aussi une force de ce modèle.[41]

3.7.2 Un peu de pratique

Dans les faits il existe deux algorithmes disponibles : VGG16 et VGG19.

Grâce à la librairie keras de Tensorflow, il est simple de récupérer le modèle déjà entraîné, par défaut, sur ImageNet.

Dans un premier temps, nous avons besoin d'appliquer le même traitement spécifique que celui appliqué au moment de l'entraînement du modèle. En plus de cela nous ajoutons de l'augmentation de données sur les données d'apprentissage pour prévenir un risque de sur-

apprentissage. Il est aussi important de vérifier que les images données en input sont des images RGB de taille 224×224.

Le principe est très simple, l'idée est de décliner une image en faisant des décalages, rotations, zooms afin de la dupliquer en plusieurs exemplaires. A partir d'une image on peut donc avoir x nouvelles images et donc améliorer l'apprentissage de notre modèle par cette méthode.

Avec TensorFlow nous utiliserons ce que l'on appelle des Generator. il en existe plusieurs mais ici nous allons utiliser le Générateur d'image ImageDataGenerator() qui va faire tout ce travail de duplication pour nous et de manière automatique.

Pour commencer on configure la manière dont on va créer les déclinaisons d'image avec la fonction ImageDataGenerator()

Ensuite on va utiliser et appliquer ce Generator a nos deux jeux de données (Training et Test).[41]

```
from keras.applications.vgg16 import preprocess_input
from keras.preprocessing.image import ImageDataGenerator
train_data_generator = ImageDataGenerator(
    preprocessing_function = preprocess_input,
    # data augmentation
    rotation_range = 10,
    width_shift_range = 0.1,
    height_shift_range = 0.1,
    zoom_range = 1.1,
    horizontal_flip = True)
test_data_generator = ImageDataGenerator(
    preprocessing_function = preprocess_input)
```

Nous pouvons alors récupérer les poids optimisés issus des couches de convolution et entraîner les 3 couches Dense que nous ajoutons et compiler.

```
import keras
from keras.layers import Dense, Flatten
from keras.optimizers import Adam
from keras.models import Model, Sequential
from keras import backend as K
from keras.applications.vgg16 import VGG16
# Modèle VGG16
base_model = VGG16(weights='imagenet', include_top=False)
# Freezer les couches du VGG16
for layer in base_model.layers:
    layer.trainable = False
model = Sequential()
model.add(base_model) # Ajout du modèle VGG16
model.add(Flatten())
model.add(Dense(4096, activation='relu'))
model.add(Dense(4096, activation='relu'))
model.add(Dense(n_class, activation='softmax'))
model.compile(optimizer='adam', loss='sparse_categorical_crossentropy',
metrics=['acc'])
```

3.8 Gradio

Permet au Data Scientist de gagnés du temps à créer des interfaces pour interagir avec leur modèles de Machine learning ? Gradio est alors la bonne solution pour nous permettre de créer des formulaires en toute simplicité.

En peut alors publier ces formulaires de saisie (et visualisation de résultat) et même récupérer des informations.

3.8.1 Installation

L'installation est très simple. un simple pip suffira ...

- 1- Pip install gradio

Note: si en utilise Google colab, et bien il suffira de placer la commande pip ci-dessus directement dans une cellule (la première) du notebook.

Un fois installé, il faut bien sur importer la librairie Gradio :[45]

- 1- *Import gradio as gr*

4 Description de CNN:

La convolution est l'acte consistant à prendre les données d'origine et à en créer des cartes. Le polissage est un sous-échantillonnage, le plus souvent sous la forme de "pooling maximal", où nous sélectionnons une région, puis prenons la valeur maximale dans cette région. , et cela devient la nouvelle valeur pour toute la région. Les couches entièrement connectées sont des réseaux de neurones typiques, dans lesquels tous les nœuds sont "entièrement connectés".

Les couches convolutives ne sont pas entièrement connectées comme un réseau de neurones traditionnel. La structure CNN de base est la suivante: Convolution -> Pooling -> Convolution -> Pooling -> Couche entièrement connectée -> Sortie.[49]

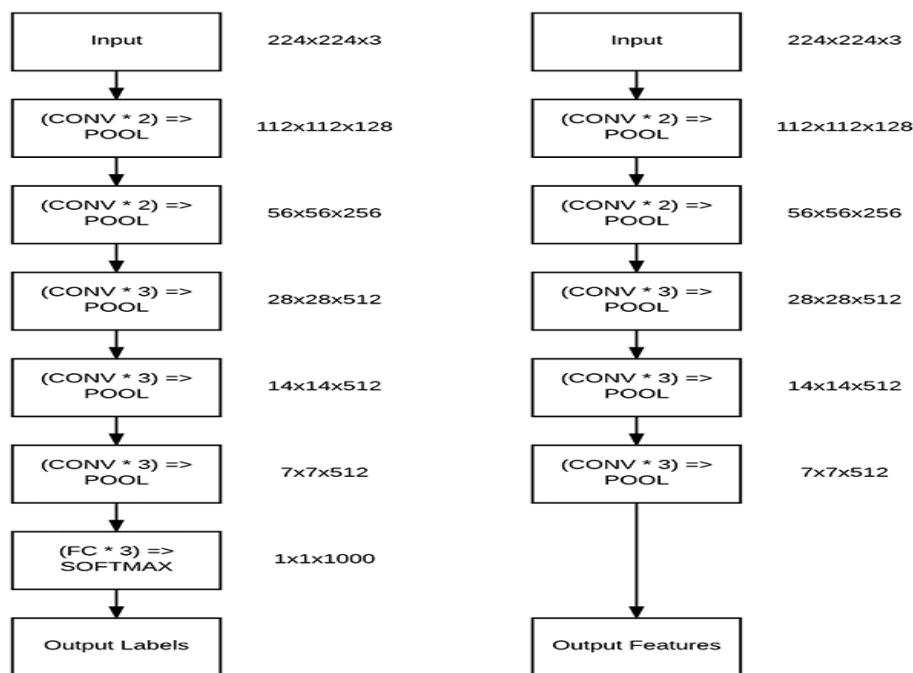


Figure 4.7: la structure CNN

5 L'architecture de l'application

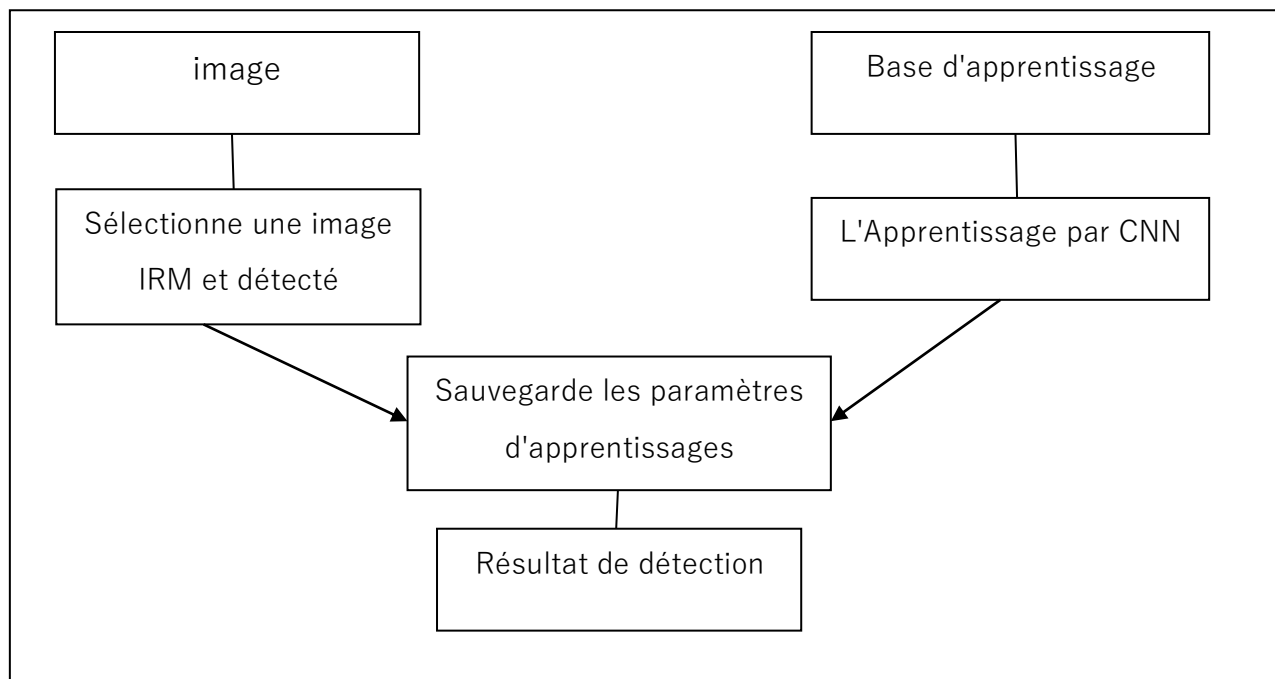


Figure 4.8: Architecteur de l'application

6 Base d'apprentissage :

Dans cette étape, on dispose les images pour l'apprentissage. Nous avons utilisé quatre classes. Le tableau ci-dessous illustre la structure de notre base d'apprentissage et les classes utilisées :

Classe	Apprentissage	Test
Normal	10192 images	10192 images
covid	3616 images	3616 images
veral-Pneumonia	1345 images	1345 images
Lung-opacity	6012 images	6012 images

7 L'apprentissage par CNN :

Dans cette étape les images de l'apprentissage sont l'entrée de CNN.

La structure de CNN utilisée est illustrée dans la figure 4.9.

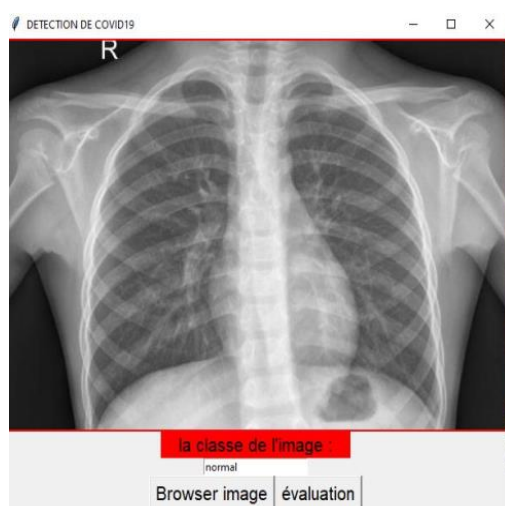


Figure 4.9 : Apprentissage Image

7.1 Sélection des images IRM :

On sélectionne un ensemble des images IRM

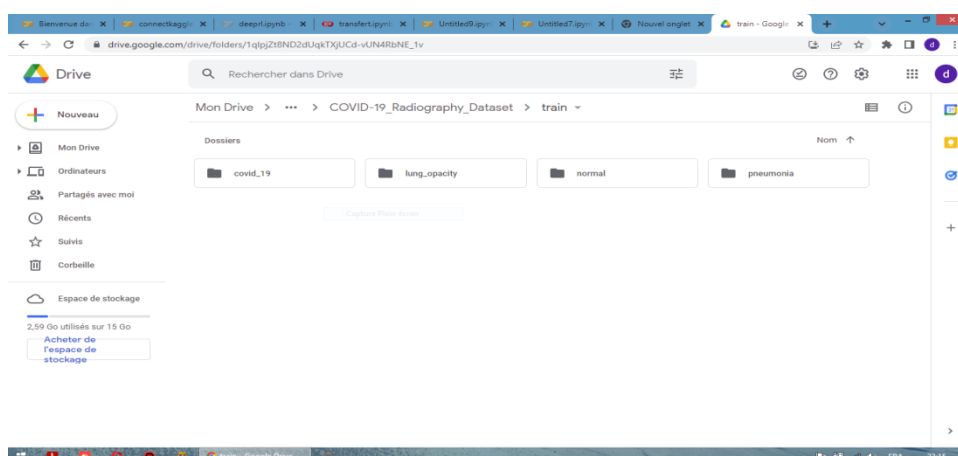


Figure 4.10 : sélection un fichier des images IRM

7.2 Sauvegarder les paramètres de l'apprentissage :

Après l'étape de l'apprentissage les paramètres sont sauvegardés, Ces paramètres sont utilisés pour le test et la détection.

7.3 La base de données :

La base de données des images IRM (Pneumonia) contient 21165 images IRM divisée en 4 classes (viral Pneumonia, Normal, covid et lung opacity).



Figure 4.11: Exemple d'image IRM de la base Pneumonia

7.4 Résultat et discussion

On a réalisés trois applications de deep learning qui contient: les modèles **CNN** , **VGG16** et **RESNET50**,Et trois applications de la apprentissage par renforcement avec les modèles des agents **PPO**, **PPO2** et **DQN** .

7.4.1 partie deep learning

La figure suivante montre le résultat de test obtenu par notre application :

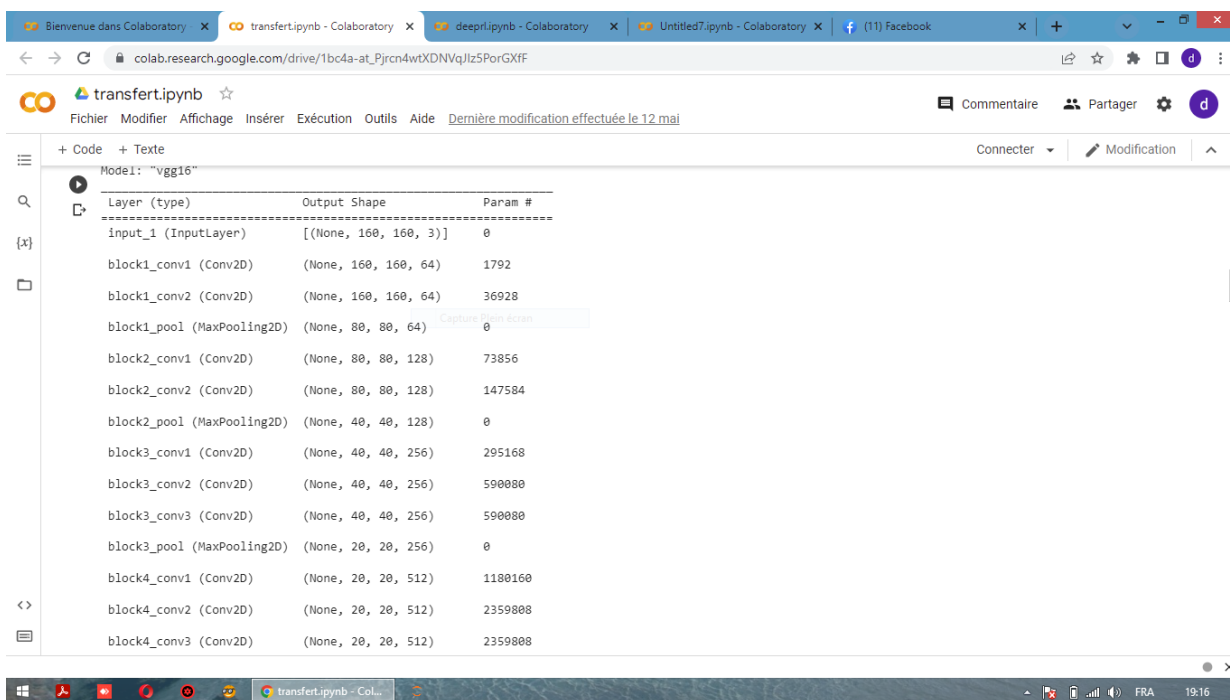
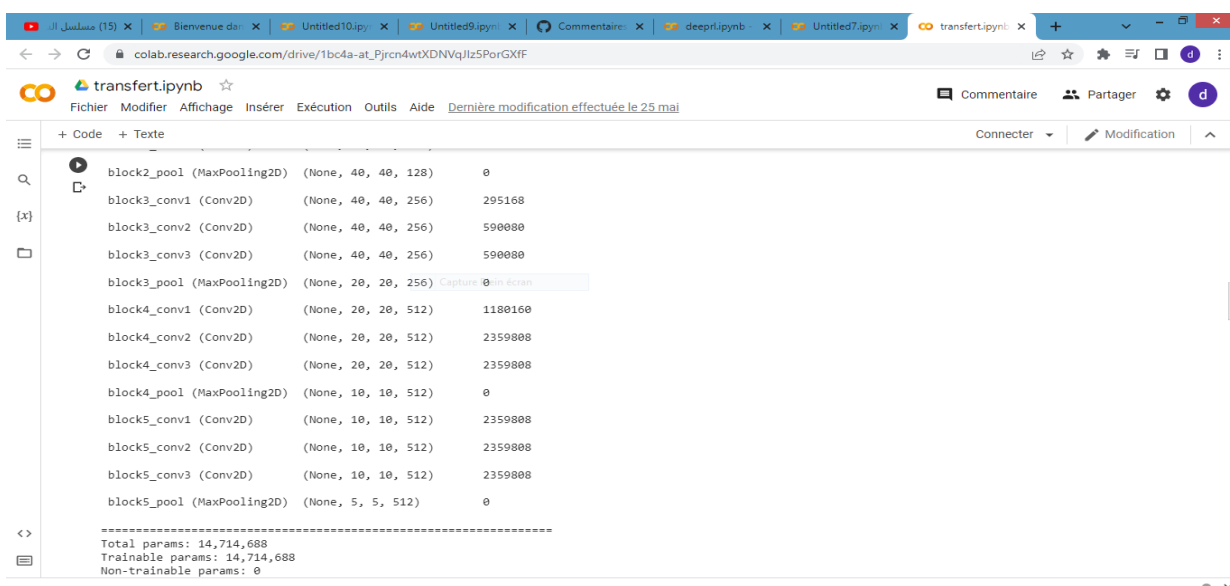


Figure 4.12: le résultat de prédiction



Suite de la figure 4.12

Pour montrer les résultats obtenus à partir de notre modèle, nous allons les préciser dans les résultats suivants en termes de précision et d'erreur (précision/perte), En plus de la matrice de confusion

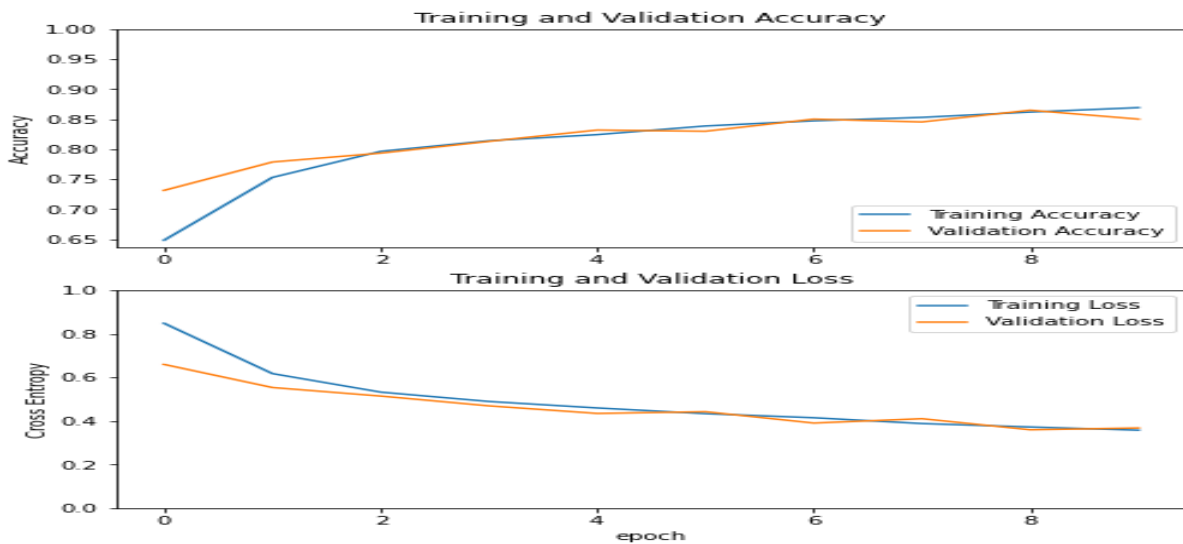


Figure 4.13: montre l' Accuracy et Loss

Le graph ci-dessus du modèle CNN illustre que la phase d'apprentissage et la phase de test ils prennent la même direction, ce qui indique le meilleur fonctionnement avec meilleure performance , et que le modèle est bien entraîné.

7.5 Matrice de confusion :

Qu'est ce qu'une matrice de confusion ?

Dans les problèmes de classification, il prédit des résultats que l'on doit comparer à la réalité pour mesurer son degré de performance. On utilise généralement la matrice de confusion, appelée aussi tableau de contingence. Elle mettra non seulement en valeur les prédictions correctes et incorrectes mais nous donnera surtout un indice sur le type d'erreurs commises [44]. Chaque colonne du tableau contient une classe prédite par l'algorithme et les lignes des classes réelles [44].

On classe les résultats en 4 catégories [44] :

- True Positive (TP) : la prédiction et la valeur réelle sont positives.

Exemple : Une personne malade et prévu malade.

- True Négative (TN) : la prédiction et la valeur réelle sont négatives.

Exemple : Une personne saine et prévu saine.

- False Positive (FP) : la prédiction est positive alors que la valeur réelle est négative.

Exemple : Une personne saine et prévu malade.

- False Négative (FN) : la prédiction est négative alors que la valeur réelle est négative.

Exemple : Une personne malade et prévu saine.

		Predicted values	
		Positive	Negative
Actual values	Positive	True Positive (TP)	Fals Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

*A partir de la matrice de confusion on peut dériver tout un tas de critères de performance. Voici quelques exemples de mesures de performance souvent utilisées : Le **rappel** ("recall" en anglais), ou sensibilité ("sensitivity" en anglais), est le taux de vrais positifs, c'est à dire la proportion de positifs que l'on a correctement identifiés. C'est la capacité de notre modèle à détecter tous les incendies [44].

$$\mathbf{Rappel} = \frac{TP}{TP+FN}$$

On s'intéressera donc aussi à la précision, c'est-à-dire la proportion de prédictions correctes parmi les points que l'on a prédits positifs. C'est la capacité de notre modèle à ne déclencher d'alarme que pour un vrai incendie. [44].

$$\mathbf{Précision} = \frac{TP}{TP+FP}$$

Pour évaluer un compromis entre rappel et précision, on peut calculer la "Fmesure", qui est leur moyenne harmonique

$$\mathbf{F - mesure} = 2 * \frac{\text{Précision} * \text{Rappel}}{\text{Précision} + \text{Rappel}} = 2 * \frac{2TP}{2TP + FP + FN}$$

Accuracy : permet de connaître la proportion de bonnes classifications par rapport à toutes les prédictions.

$$\mathbf{Accuracy} = (TN+TP)/(TP+TN+ FP+FN)$$

1- Matrice de confusion du modèle CNN

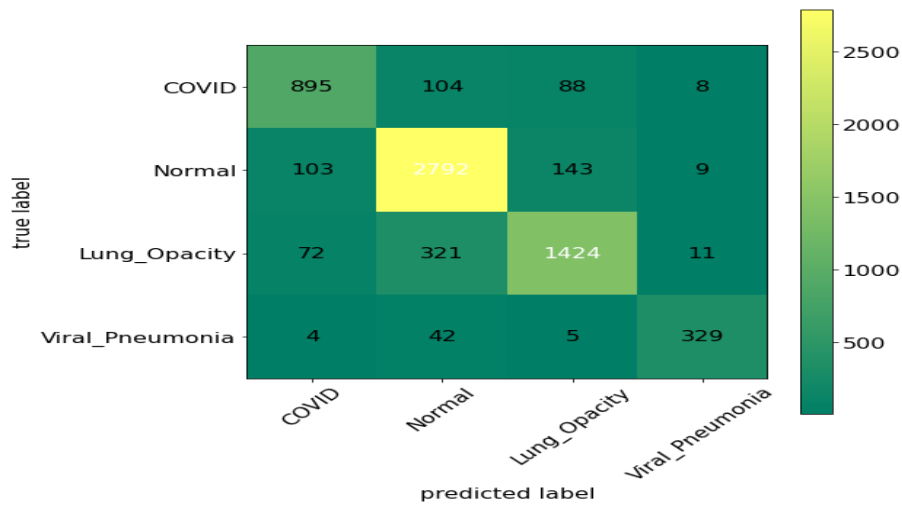


Figure 4.14: matrice de confusion du modèle CNN

En constate que la meilleure classe dans la matrice de confusion c'est la classe NORMALE.

La accuracy = **0,86** dans le modèle CNN comme la montre la figure suivante.

	precision	recall	f1-score	support
COVID	0.83	0.82	0.83	1095
Normal	0.86	0.92	0.89	3047
Lung_Opacity	0.86	0.78	0.82	1828
Viral_Pneumonia	0.92	0.87	0.89	380
accuracy			0.86	6350
macro avg	0.87	0.84	0.86	6350
weighted avg	0.86	0.86	0.86	6350

2- Matrice de confusion du modèle VGG16

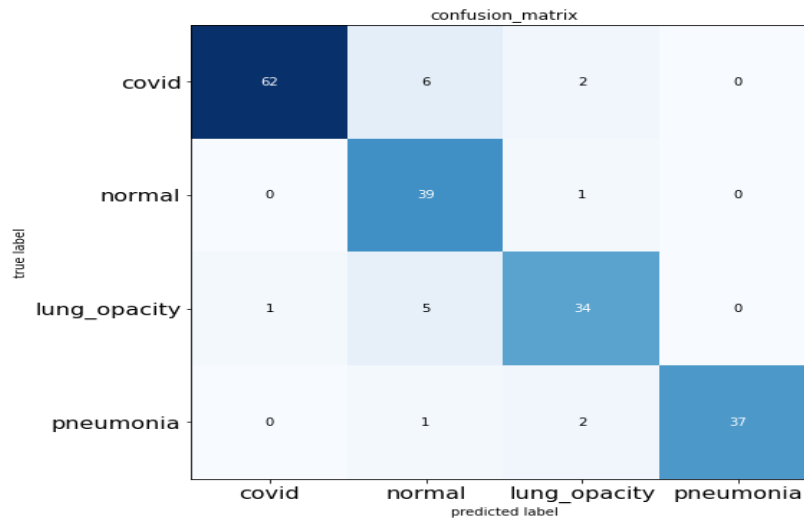


Figure 4.15: matrice de confusion modèle vgg16

La figure ci-dessus montre la matrice de confusion du modèle vgg16 qui illustre que la classe covid est la meilleure classe.

On a trouvé ACCURACY = **0,91**, résultat du modèle vgg 16 meilleur que la matrice de confusion du modèle CNN.

```
confusion_matrix
```

```
[[62 6 2 0]
```

```
[ 0 39 1 0]
```

```
[ 1 5 34 0]
```

```
[ 0 1 2 37]]
```

```
classification report
```

	precision	recall	f1-score	support
covid	0.98	0.89	0.93	70
normal	0.76	0.97	0.86	40
lung_opacity	0.87	0.85	0.86	40
pneumonia	1.00	0.93	0.96	40
accuracy			0.91	190
macro avg	0.91	0.91	0.90	190
weighted avg	0.92	0.91	0.91	190

é

La figure suivante montre le training du model vgg16 qui donne les meilleur résultat

```

result = model.fit_generator(train_generator,
Epoch 1/10
89/89 [-----] - 713s 8s/step - loss: 1.1109 - accuracy: 0.6100 - val_loss: 0.9252 - val_accuracy: 0.64
00
Epoch 2/10
89/89 [-----] - 692s 8s/step - loss: 0.6020 - accuracy: 0.7876 - val_loss: 0.4038 - val_accuracy: 0.85
50
Epoch 3/10
89/89 [-----] - 683s 8s/step - loss: 0.4069 - accuracy: 0.8449 - val_loss: 0.2639 - val_accuracy: 0.92
00
Epoch 4/10
89/89 [-----] - 789s 9s/step - loss: 0.3279 - accuracy: 0.8742 - val_loss: 0.2737 - val_accuracy: 0.88
00
Epoch 5/10
89/89 [-----] - 786s 9s/step - loss: 0.2501 - accuracy: 0.9169 - val_loss: 0.3978 - val_accuracy: 0.82
00
Epoch 6/10
89/89 [-----] - 790s 9s/step - loss: 0.2506 - accuracy: 0.9135 - val_loss: 0.3173 - val_accuracy: 0.83
50
Epoch 7/10
89/89 [-----] - 785s 9s/step - loss: 0.2191 - accuracy: 0.9247 - val_loss: 0.1825 - val_accuracy: 0.94
50
Epoch 8/10
89/89 [-----] - 786s 9s/step - loss: 0.1925 - accuracy: 0.9292 - val_loss: 0.3231 - val_accuracy: 0.84
50
Epoch 9/10
89/89 [-----] - 788s 9s/step - loss: 0.1947 - accuracy: 0.9303 - val_loss: 0.1784 - val_accuracy: 0.90
50
Epoch 10/10
89/89 [-----] - 787s 9s/step - loss: 0.1733 - accuracy: 0.9382 - val_loss: 0.1733 - val_accuracy: 0.92
50
    
```

Figure 4.16: montre le résultat du training du model vgg16.

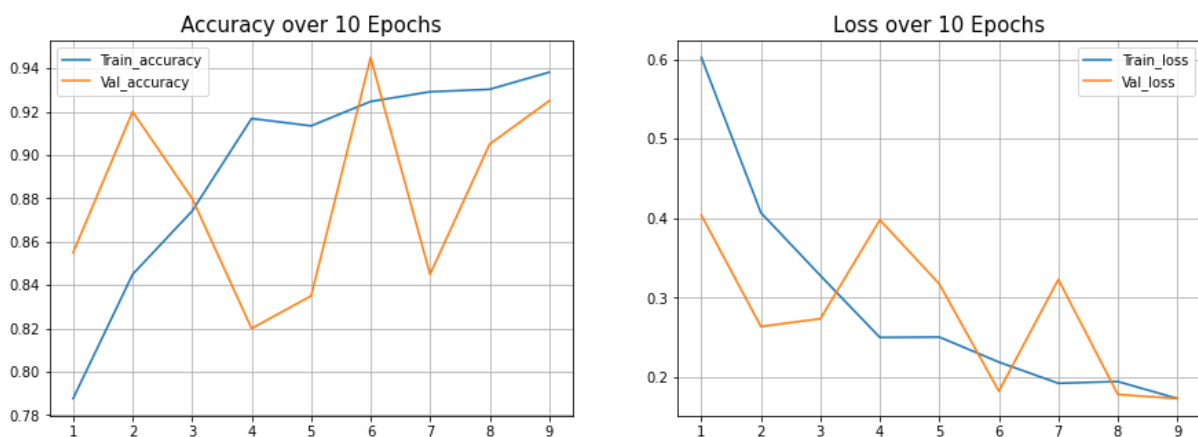


Figure 4.17: illustration de l'overfitting du model vgg16

L'overfitting est un problème qui est souvent rencontrés en machine learning. Il survient lorsque notre modèle essaye de trop coller aux données d'entraînements.

Comment savoir si un modèle fait de l'overfitting ?

En data science les données parfaites n'existent pas. Nous avons toujours du bruit et des imprécisions. Un modèle fait de l'overfitting lorsqu'il commence à apprendre ce bruit. Il en résulte un modèle biaisé qui est impossible à généraliser.

En pratique, un modèle qui overfit est souvent très facile à détecter. L'overfitting intervient lorsque l'erreur sur les données de test devient croissante. Typiquement, si l'erreur sur les données d'entraînements est beaucoup plus faible que celle sur les données de test, c'est sans doute que notre modèle a trop appris les données.

Comment éviter l'overfitting ?

Plusieurs techniques existent pour éviter l'overfitting. Parmi les méthodes les plus utilisées.

1- Cross-validation

Une des méthodes les plus efficaces pour éviter l'overfitting est le *cross validation* (validation croisée en français).

En cross validation on divise les données d'entraînements en plusieurs groupes. L'idée est ensuite d'entraîner le modèle sur tous les groupes sauf un. Si on a k groupes, on entraînera le modèle k fois avec à chaque fois un nouveau groupe de test.

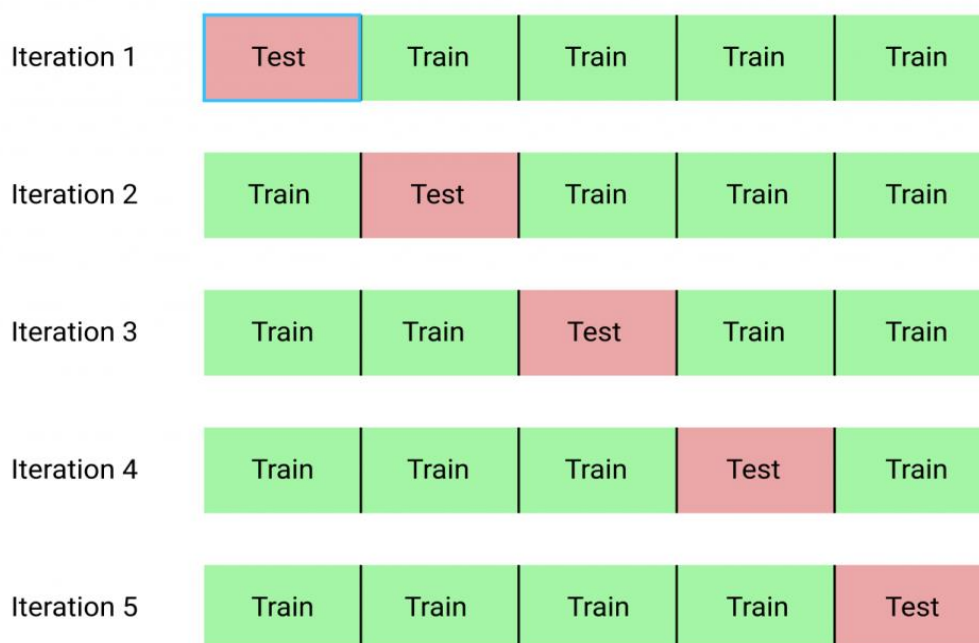


Figure 4.18: Cross validation

3- Matrice de confusion du modèle resnet50

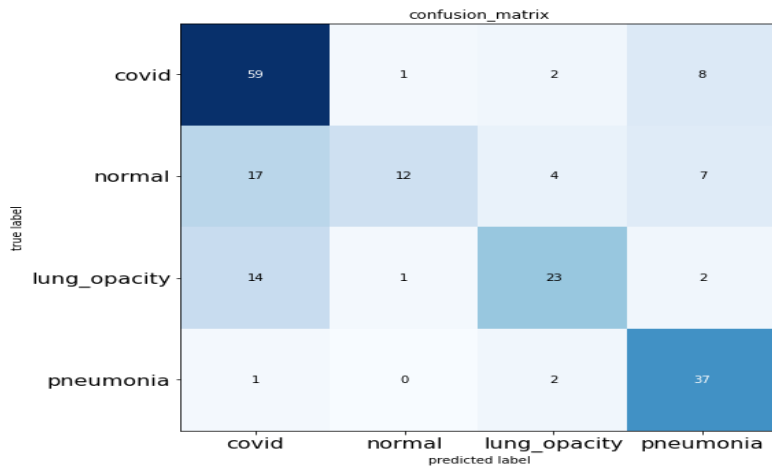


Figure 4.19: matrice du modèle ResNet50

La figure ci-dessus montre la matrice de confusion du modèle ResNet50 qui illustre que la classe covid est la meilleure classe et que l'Accuracy = 0.69.

confusion_matrix

```
[[59 1 2 8]
 [17 12 4 7]
 [14 1 23 2]
 [ 1 0 2 37]]
```

classification report

	precision	recall	f1-score	support
covid	0.65	0.84	0.73	70
normal	0.86	0.30	0.44	40
lung_opacity	0.74	0.57	0.65	40
pneumonia	0.69	0.93	0.79	40
accuracy			0.69	190
macro avg	0.73	0.66	0.65	190
weighted avg	0.72	0.69	0.67	190

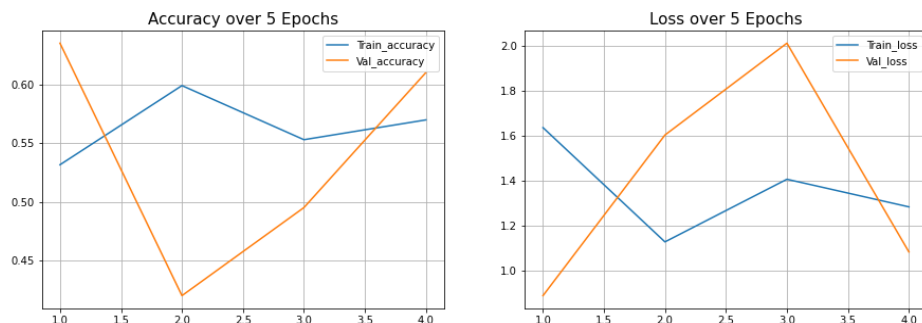


Figure 4.20: illustration de l'overfitting du model ResNet50

étude comparative entre les 3 modèles(CNN, VGG16 et Resnet50)

➤ **Coté de paramètres :**

modèles	Total params	Trainable params	Non-trainable params
CNN	108,628	108,628	0
VGG16	14,815,044	100,356	14,714,688
ResNet 50	23,989,124	401,412	23,587,712

Le tableau ci-dessus montre qu'il y a une différence entre les paramètres. Et le modèle Resnet50 C'est le modèle qui contient plus de paramètres que les autres modèles.

➤ **Coté train modèles**

1- modèle CNN

Les résultats du modèles CNN au niveau de l'accuracy finale est 0,85 et le taux erreurs diminue de 0,65 jusqu'à 0,35 coté de formation et dans l'autre coté validation elles sont les mêmes donc le modèle est bien formé et apte à utiliser.

2- modèle VGG16

On a le nombre d'epoch c'est le nombre de formation est qui = 10

Les résultats obtenue au niveau de l'ACCURACY =0,93 et le taux erreur = (1,11 jusqu'à 0,17) c'est la partie de formation.

Dans la partie validation ACCURACY=0,92 et le taux erreur (0,64 jusqu'à 0,92)

Donc on conclu que le modèle VGG16 au niveau exactitude meilleure que CNN mais dans loss CNN mieux que VGG16.

3- modèle ResNet50

On a ACCURACY =0,56 et LOSS =1,38 jusqu'à 1,28 dans train et dans la validation ACCURACY =0,61 et LOSS =2,26 jusqu'à 1,08

Après l'analyse des résultats obtenus, On constate les remarques suivantes : qu'on a l'overfitting dans les modèles VGG16 et RESNET50 et que le modèle CNN valide avec meilleure performance. Et que la meilleure classe parmi les 3 modèles dans les matrices de confusions c'est la classe covid, Et meilleure accuracy c'est l'accuracy du modèle VGG16 qui atteint 0.91.

8 partie de l'Apprentissage par renforcement(AR)

L'apprentissage par renforcement est une approche informatique de l'apprentissage dans laquelle un agent essaie de maximiser le montant total de la récompense qu'il reçoit en interagissant avec un environnement complexe et incertain. L'objectif de l'agent est d'apprendre, à partir d'expériences, ce qu'il convient de faire en différentes situations. Les récompenses numériques reçues pour les essais(essai / l'erreur) sont utilisées dans ce but. Plusieurs algorithmes existent pour effectuer cet apprentissage ; ils se basent tous sur une représentation du problème sous forme d'un processus décisionnel de Markov .

8.1 Résultat et discussion :

On a utilisé trois modèle sont les suivants :

- 1 - PPO
- 2 - PPO2
- 3 - DQN

1) modèle de rl Agent ppo

rollout/	
ep_len_mean	1
ep_rew_mean	0.73
time/	
fps	399
iterations	20
time_elapsed	102
total_timesteps	40960
train/	
approx_kl	0.05058814
clip_fraction	0.185
clip_range	0.2
entropy_loss	-0.268
explained_variance	0.166
learning_rate	0.0003
loss	0.0467
n_updates	190
policy_gradient_loss	0.0254
value_loss	0.165

Ce tableau contient les résultats du modèle rl d'agent ppo

Discussion du tableau

les variables qui contiennent ce tableau sont:

1-rollout : indique la sortie.

1-1- ep_len_mean: durée moyenne de l'épisode.

1-2- ep_rew_mean : signifie la récompense de l'épisode.

2 temps.

2-1 –total timesteps : nombre de pas de temps(c'est-à-dire nombre d'action effectuées dans ENV).

3- train :

3-1-approxkl : mesure approximative de l'ancienne politique par rapport à la nouvelle politique.

3-2-clip frac :fraction de fois on hyper paramètre de plage de découpage est utilisé.

3-3-entropy : décision de modèle diminue-> BETA hyper paramètre augmente.

3-4-learning rate: strategies optimal.

3-5-policy loss : processus decision des actions.

3-6-value loss : la perte moyenne de la mise à jour de la fonction de valeur.

3-7-explained_variance :c'est $ev > 0$ bon indicateur du rendement c'est $ev < 0$ rien prédire.

3-8-policy gradient loss :fonction de récompense.

le graphe de ppo

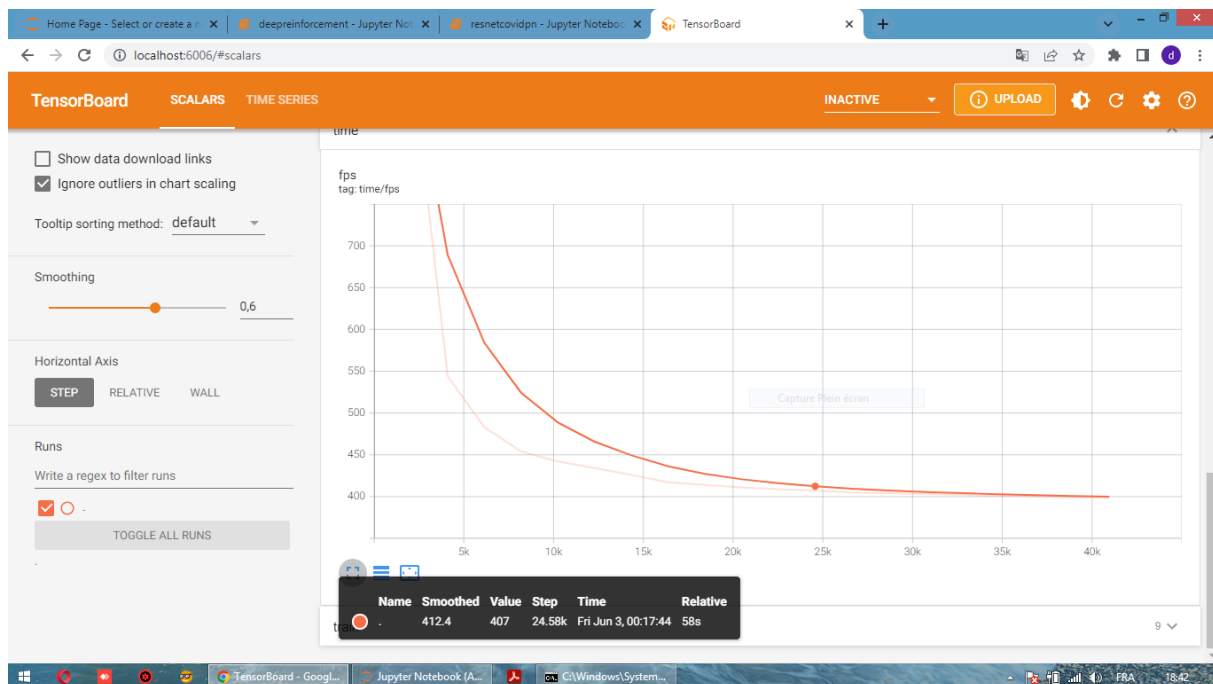


Figure 4.21: le nombre de pas de temps du modèle ppo

Figure ci-dessus montre le nombre de pas de temps (c'est-à-dire nombre d'action effectuées dans ENV)

Le graph ci- dessus affiche la durée moyenne de l'épisode ep-len-mean

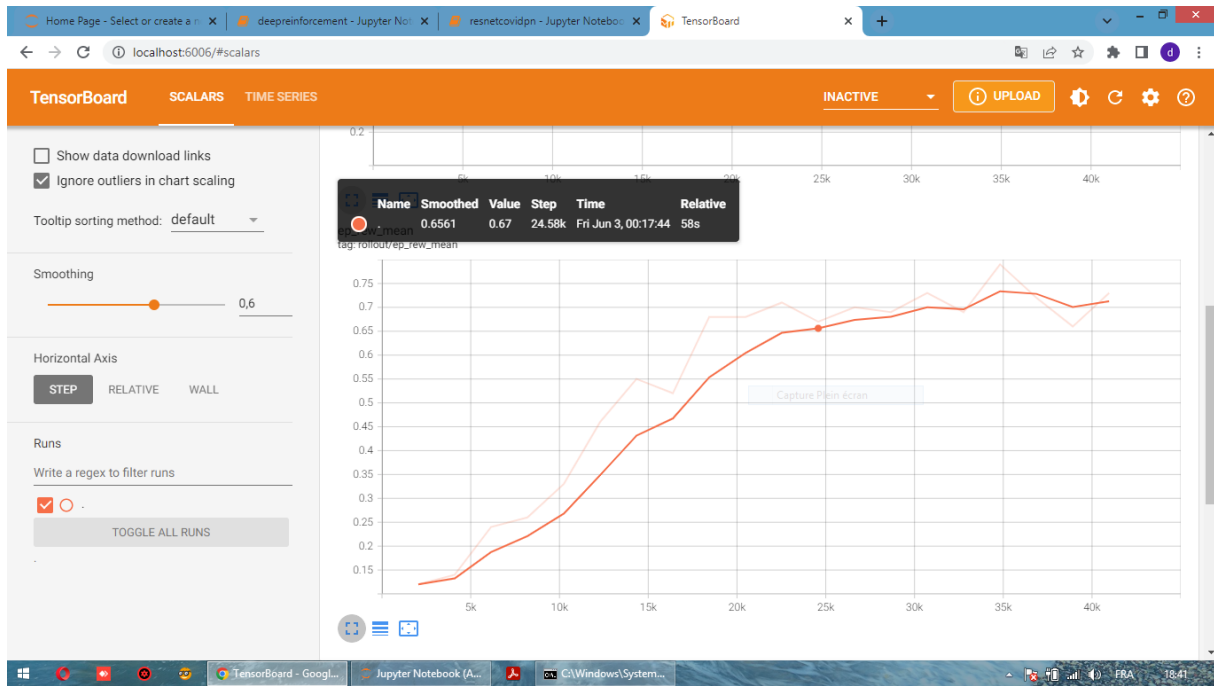


Figure 4.22: le nombre de pas de temps du modèle ppo

Le graphe ci-dessous montre la récompense de l'épisode (Ep-rew-mean)

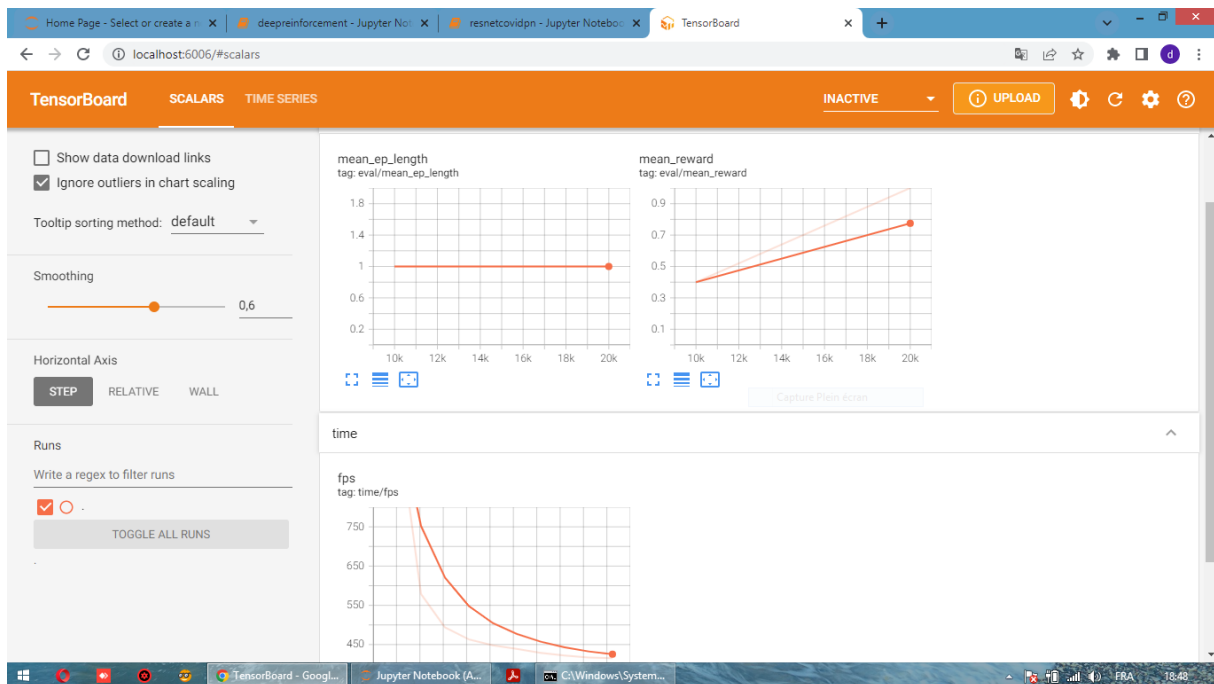


Figure 4.23: montre la récompense d'épisode du modèle ppo

2) modèle DQN

Tableau suivant contient les résultats rl du modèle d'agent **DQN**.

IL contient les variables de temps et résultat de la récompense dans 100 épisodes

```
-----
| % time spent exploring | 1 |
| episodes               | 1.2e+05 |
| mean 100 episode reward | 0.8 |
| steps                  | 1.2e+05 |
-----
```

```
-----
| % time spent exploring | 1 |
| episodes               | 1.2e+05 |
| mean 100 episode reward | 0.8 |
| steps                  | 1.2e+05 |
-----
```

Restored model with mean reward: 0.8
DQN Training Time: 9657.364847660065

Le graph de la figure 4.24 du modèle DQN montre le résultat de la récompense par apport à l'épisode et le temps.

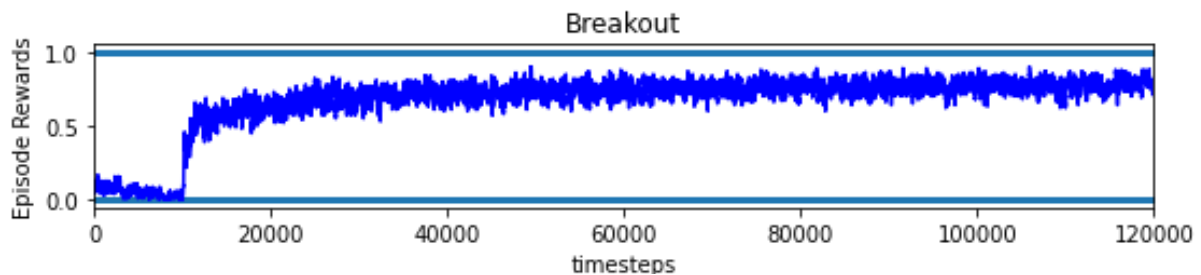


Figure 4.24: montre la récompense d'épisode du modèle DQN

Le graph ci-dessus montre les résultats en termes de récompense. Nous remarquons dans la figure que l'agent a commencé à s'entraîner lourdement mais avec le temps à commencé à nous donner de bons résultats.

3) modèle PPO2

```
-----
| eplenmean              | 1 |
-----
```

```

| eprewmean           | 0.59 |
| fps                 | 26   |
| loss/approxkl      | 0    |
| loss/clipfrac      | 0    |
| loss/policy_entropy | 0.000185 |
| loss/policy_loss   | -9.55e-09 |
| loss/value_loss    | 0.124 |
| misc/explained_variance | 0    |
| misc/nupdates      | 340  |
| misc/serial_timesteps | 4.35e+04 |
| misc/time_elapsed  | 1.61e+03 |
| misc/total_timesteps | 4.35e+04 |

```

ce tableau contient les résultats du rl du modèle d'agent PPO2.

résultat reward=0,59.

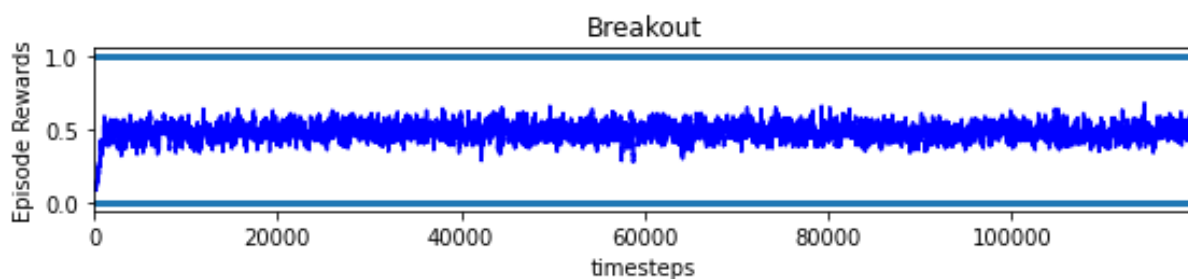


Figure 4.25: montre la récompense d'épisode du modèle PPO2

Graph de PPO2 montre le résultat de la récompense par apport à l'épisode et le temps.

Après cette expérience nous avons conclu que le modèle PPO à donner une reward =0,8.

Ce que veut dire le bon fonctionnement de l'agent. Ainsi que le modèle DQN nous a donné des bons résultats pour la classification d'images dans le feature.

9 L'interface de l'application :

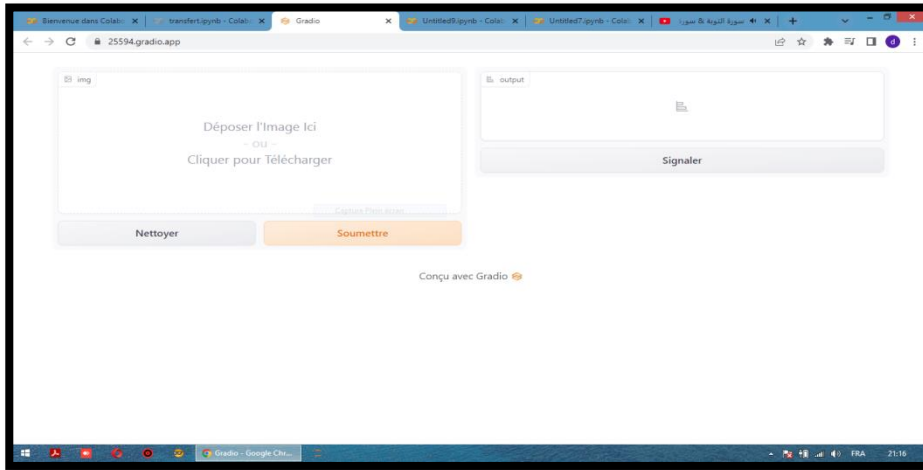


Figure 4.26: l'interface de l'application

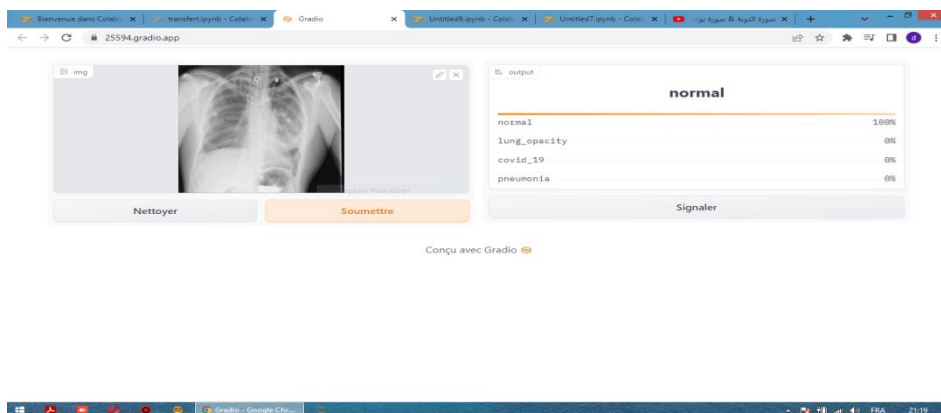


Figure 4.27 : exemple d'image IRM de la base normal l'interface de l'application

Sur le côté gauche de l'écran, nous plaçons l'image sélectionnée, et sur le côté droit de l'écran, les résultats sont affichés en pourcentage.

10 La comparaison de résultat avec d'autre travail :

Nous comparons notre résultat avec le travail «*Détection automatique du Covid-19 sur les images radiographiques en utilisant le deep learning par le réseau CNN*»

Le résultat de notre travail avec RL a donné un excellent résultat en s'entraînant en temps et en rapidité qui atteignant un résultat de 80 % (précise sans erreurs). Contrairement au deep learning qui demande beaucoup de temps et d'époque.

La rapidité de l'apprentissage peut être atteinte par une technique d'apprentissage par renforcement indirect. Le système apprend un modèle de l'environnement à partir d'interactions réelles et utilise ce modèle pour simuler des expériences supplémentaires et ainsi apprendre plus rapidement. Atteignant

Nous avons comparé les modèles que nous avons utilisés entre eux et dans le tableau ci-dessous nous pouvons voir le résultat obtenu.

1) La différence entre les modèles du deep learning qu'on a utilisés par apport au résultat:

modèles	accuracy	loss	val accuracy	val loss
CNN	0.86	0.35	0.85	0.36
VGG16	0.93	0.17	0.92	0.17
ResNet50	0.56	1.28	0.61	1.08

on remarque dans le tableau ci-dessus que le modèle VGG16 c'est le meilleur modèle accuracy=0.93

2) Comparaison entre les modèles de l'apprentissage par renforcement:

1) Modèle DQN REWARD=

Restored model with mean reward: 0.8
DQN Training Time: 9657.364847660065

2) Modèle PPO REWARD=1 total_timesteps =20000.

3) Modèle ppo2 reward=0,59
PPO2 Training Time: 4425.813000202179

Après cette expérience, nous concluons que le modèle PPO est le meilleure modèle de Classification d'image au niveau de la précision et le temps. il prend peu de temps avec une bonne précision.

11 Conclusion :

Dans ce chapitre nous avons expliqué l'architecture de l'application (deep learning et apprentissage par renforcement), les résultats obtenus, et nous avons comparé notre résultats avec d'autre travail. Discuter et interpréter les résultats obtenus, Détailler les étapes de mise en œuvre de notre travail, Réaliser une étude comparative entre les modèles d'apprentissage, de plus, nous avons fait un comparatif entre les modèles de deep learning.

Conclusion Générale



ION
LE

Marmite Informatique

Conclusion générale :

La classification d'images est une tâche importante dans le domaine de la vision par ordinateur, la reconnaissance d'objets et l'apprentissage automatique. Grâce à l'apprentissage en profondeur (Deep Learning), l'avenir de l'intelligence artificielle dans le développement de grandes applications est très rapide.

Dans ce projet nous avons présenté une des opérations de traitement des images qui est la détection de Covid-19 en utilisant des images IRM et nous avons utilisé deux modèles. Le premier c'est l'apprentissage profond avec un réseau de neurones profond (Convolutional Neural Network - CNN) car depuis quelques années ces réseaux ont montré de bonnes performances et sont devenus l'état de l'art dans le domaine de la vision par ordinateur, surtout pour la classification d'images. Les CNN sont composés d'une succession de couches de produit de convolution des images. Ces produits de convolution servent à extraire petit à petit les informations caractéristiques sur l'image traitée. Plus spécifiquement, dans notre travail nous avons utilisé un transfert d'apprentissage. Le transfert d'apprentissage est utilisé en apprentissage profond et a pour principe de migrer les connaissances d'un modèle déjà appris (les paramètres) vers un nouveau problème. L'architecture et les paramètres du CNN sont ainsi transférés. Dans le premier modèle développé, nous avons utilisé le principe d'encodeur-décodeur qui est une stratégie utilisée pour générer automatiquement les légendes des images. Dans notre cas, cette stratégie a été appliquée pour la classification des images. Plus spécifiquement, à partir d'un CNN déjà entraîné, nous avons extrait un vecteur de caractéristiques (encodage). Après cet encodage, le vecteur de caractéristiques est décodé par un autre réseau de neurones afin d'obtenir les concepts. Le modèle CNN donne un résultat d'accuracy = 0.85 ainsi la matrice de confusion, donc le modèle a été bien entraîné mais le problème existe toujours dans le facteur du temps.

Le deuxième modèle que nous avons utilisé c'est l'Apprentissage par renforcement (AR). L'apprentissage apparaît comme une bonne approche car il permet de personnaliser l'environnement sans spécification explicite de la part de l'utilisateur. Un apprentissage à vie permet, par ailleurs, de toujours s'adapter aux modifications du monde et des préférences utilisateur.

L'apprentissage par renforcement est un paradigme d'apprentissage qui peut être une solution à notre problème, à condition de l'adapter aux contraintes liées à notre cadre d'application.

L'apprentissage par renforcement permet de remplir ces conditions et d'atteindre le but. L'entraînement est naturellement simple pour l'utilisateur puisqu'il s'agit de récompenses subjectives, positives ou négatives, reflétant la satisfaction de l'utilisateur vis-à-vis du comportement du système. La rapidité de l'apprentissage peut être atteinte par une technique d'apprentissage par renforcement indirect. Le système apprend un modèle de l'environnement à partir d'interactions réelles et utilise ce modèle pour simuler des expériences supplémentaires et ainsi apprendre plus rapidement.[50]

L'objectif de ce projet est de détecter le Covid-19 de manière rapide et précise pour réduire l'infection de la maladie. Après plusieurs tentatives, nous avons constaté que l'apprentissage par renforcement donnait de bons résultats, meilleurs et plus rapides que l'apprentissage en profondeur, atteignant 80 % (précise sans erreurs).

Bibliographie



Bibliographie

- [1] Mhammedi Aymen et Yakoub Imane "La détection de Covid-19 par l'apprentissage profonde (Deep Learning) Université Ahmed Draia – Adrar.
- [2] Matthieu Zimmer. Apprentissage par renforcement développemental. Intelligence artificielle [cs.AI]. Université de Lorraine, 2018.
- [3] <https://www.futura-sciences.com/sante/definitions/coronavirus-covid-19->
- [4] <http://github.com/ieee8023/covid-chestxay-dataset>
- [5]- www.health.gov.au/sites/default/files/documents/2020/03/coronavirus-covid-19-frequently-asked-questions.pdf.
- [6](en) François-Xavier Lescure, Lila Bouadma, Duc Nguyen et arion Parisey, « Clinical and virological data of the first cases of COVID-19 in Europe: a case series »
- [7] <https://sante.journal-des-femmes.fr/fiches-maladies/2607859-definition-signification-famille-coronavirus-origine-nom-virus-symptome-incubation-traitement-transmission/>.
- [8] Lyudmyla Kompaniyets et al. , Underlying Medical Conditions and Severe Illness Among , cdc.gov, Volume 18, 1er juillet 2021
- [9] COMMUNICABLE DISEASE THREATS REPORT ; European Centre for Disease Prevention and Control (ECDC) 4-10 July 2021.
- [10] <https://www.vidal.fr/maladies/voies-respiratoires/coronavirus-covid-19/diagnostic.html#:~:text=Actuellement%2C%20le%20dépistage%20de%20l,profondément%20dans%20les%20fosses%20nasale>
- [11] https://www.cochrane.org/fr/CD013639/INFECTN_quelle-est-la-precision-de-limagerie-thoracique-dans-le-diagnostic-de-la-covid-19
- [12] <https://docteurimago.fr/actualites/socioprofessionnel/lirm-a-clairement-place-parmi-outils-de-medecine-moderne/>
- [13] Judd E. Hollander et Brendan G. Carr, « Virtually Perfect? Telemedicine for Covid-19 », New England Journal of Medicine, 11 mars 2020.
- [14] Xiaolong Qi, Zicheng Jiang, Qian Yu et Chuxiao Shao, « Machine learning-based CT radiomics model for predicting hospital stay in patients with pneumonia associated with SARS-CoV-2 infection: A multicenter study »- 03/mars 2020
- [15] « Maladie à coronavirus 2019 (COVID-19) : OMS (consulté le 7 septembre 2021).
- [16] Par Florence Méréo Le 19 janvier 2021 à 17h41 et Modifié le 19 janvier 2021 À 20h39, « Covid-19 : Mauricette, la première vaccinée de France, a reçu sa seconde injection ».
- [17] https://fr.wikipedia.org/wiki/Pandémie_de_Covid-19_en_Algérie#cite_note-3
- [18] Leïla Beratto, « Algérie: l'Italien porteur du coronavirus localisé à Ouargla et mis en quarantaine »
- [19] <http://www.map.toulouse.archi.fr/works/panoformation/imagenum/imagenum.htm>.
- [20] <https://openclassrooms.com/fr/courses/4470531-classez-et-segmentez-des-donnees-visuelles/4470538->
- [21] <https://pixees.fr/comment-encoder-une-image-en-noir-et-blanc/>
- [22] <https://www.youtube.com/watch?v=pMWkTS9Rpm0>, 10.nov.2020
- [23] <https://irem.univ-reunion.fr/spip.php?article903#codageCoul>
- [24] <https://www.cea.fr/comprendre/Pages/sante-sciences-du-vivant/essentiel-sur-imagerie-medicale.aspx>

- [25] <https://www.futura-sciences.com/sante/definitions/medecine-irm-8096/>
- [26] Mr. **MHAMMEDI Aymen** et **YAKOB Imane**, 2020. "la détection de covid-19 par l'apprentissage profond (Deep Learning) ": mémoire de projet de fin d'étude. Université Ahmed Draia Adrar.
- [27] Charles .Qi. CS468 Spring 2017. *Introduction to Deep Learning*.pp.2.4
- [28] <https://www.coursehero.com/file/p585s6n/Deep-learning-is-a-function-of>
- [29] **Boughaba Mohammed** et **Boukhris Brahim** " L'apprentissage profond (Deep Learning) pour la classification et la recherche d'images par le contenu" université KASDI MERBAH OUARGLA.
- [30]<https://towardsdatascience.com/covolutional-neuralnetwork>
- [31] <https://i.stack.imgur.com/mkjZo.jpg>
- [32] A. **HABBA** et O. **ISHAK**. Année Universitaire 2018/2019. *La classification des images satellitaires par l'apprentissage profonde (deep learning)*. Mémoire de fin d'étude. Université Ahmed Draia – Adrar.
- [33] T.**STÉRIN**. Stage 2015–2016. *Réseaux de neurones récurrents et mémoire : application à la musique*. Licence d'informatique. Université Claude Bernard Lyon.
- [34] Z.**Sellami**. 7 juillet 2019. *Optimisation du CNN par l'algorithme Génétique pour la Reconnaissance de Visage*. Mémoire de fin d'étude. Université Mohamed Khider de Biskra.
- [35] **Dave Gershgor**n, « The data that transformed AI research—and possibly the world »
- [36] K. **BENHAMZA** , H. **SERIDI** et H. **AKDAG** : "Conception d'Agents Intelligents avec Apprentissage par Renforcement" Département d'informatique, Université de Guelma
- [37] **Richard Sutton** and **Andrew Barto**. *Reinforcement Learning*. MIT Press, 1998.
- [38] **Tambet Matiisen**, « *Demystifying Deep Reinforcement Learning | Computational Neuroscience Lab* »
- [39] **C. J. Watkins**, *Learning from delayed rewards*, Kings College, Cambridge, mai 1989
- [40] **George F Luger**, *Artificial intelligence : Structures and strategies for complex problem solving*. 5e édition.
- [41] <https://datascientest.com/quest-ce-que-le-modele-vgg>.
- [42]<https://techterms.com/definition/python>
- [43]<http://weegee.vision.ucmerced.edu/datasets/landuse.html>
- [44] **SAIDI BEN BRAHIM** et **BAKOU RAMDANE** *Détection automatique du Covid-19 sur les images radiographiques en utilisant le deep learning par le réseau CNN(université de saida)*
- [45]<http://www.datacorner.fr/gradio/>"datacorner par Benoit Cayla"
- [46] **ZERROUGUI, Mounir**; **HAMADENE, Soumaya**"Détection de la tumeur cérébrales dans l'image IRM par l'Apprentissage en profondeur" univertsité El Bachir El Ibrahimy Bordj Bou Arrirdj.
- [47] **Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li et Li Fei-Fei**, 2009 conference on *Computer Vision and Pattern Recognition*, 2009.
- [48] **Zhang, Ren, Sun** (2015) *Deep Residual Learning for Image Recognition*
- [49] <https://www.pyimagesearch.com/2019/05/27/keras-feature-extraction-on-largedatatasetswith-deep-Learning/>
- [50] **Sofia ZAIDENBERG** " Apprentissage par renforcement de modèles de contexte pour l'informatique ambiante" , l'École Doctorale Mathématiques, Sciences et Technologies de l'Information, Informatique Grenoble.