



République Algérienne Démocratique et Populaire
Ministère de l'enseignement supérieur et de la
recherche scientifique



Université Larbi Tébessi - Tébessa
Faculté des Sciences Exactes et des Sciences de la
Nature et de la vie
Département : Mathématique et informatique

**Mémoire de fin d'étude
Pour l'obtention du diplôme de MASTER**

Domaine : Mathématiques et informatique
Filière : Informatique
Option : Système et multimédia

Thème

**DEEP LEARNING APPROCHE POUR
IDENTIFIER/DETECTION COVID19
BASSE (IMAGES X-RAY)**

Présenté Par :

Fatma Zohra Boukhalfa

Devant le jury :

| | | | |
|-------------------|-----|--------------------------|-----------|
| Dr.Djeddi Chawki | MCA | Université Larbi Tébessi | Président |
| Mr.Hajaj Ismail | MAA | Université Larbi Tébessi | Examineur |
| Dr.Merzoug Soltan | MCB | Université Larbi Tébessi | Encadreur |

Date de soutenance :22 juin 2021

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Remerciements

Nous remercions ALLAH le tout puissant pour nous avoir donné la foi et éclairé notre chemin vers la réussite durant toutes nos années d'étude.

*Nous voudrions à remercier notre encadreur de mémoire **Mr.Merzoug***

Softan qui a accepté la direction de ce travail ainsi que pour ses conseil judicieux et précieux, ses compétences scientifiques, sa confiance qu'elle nous a accordé et surtout pour ses très grands qualités humaines et sa gentillesse.

Nous remercions vivement les membres de ce jury :

- **Mr.Djeddi Chawki**, nous sommes très honorés que vous avez accepté la présidence du jury de ce mémoire. Trouvez ici l'expression de nos sincères remerciements et soyez assurée de notre profonde gratitude.*

- **Mr.Hajaj Ismaïl**, merci d'avoir accepté de faire partie du jury de ce mémoire, pour l'intérêt que vous portez à notre travail*

Enfin nous remercions toutes les personnes qui nous ont encouragé et soutenu de près ou de loin durant la réalisation de ce travail.

Dédicace

A ma mère & mon père , aucune dédicace ne saurait exprimer mon respect, mon amour éternel et ma considération pour les sacrifices que vous avez consenti pour mon instruction.

A ma sœur & mes frères , pour l'amour qu'elles me réservent et qui ont toujours été présentées dans tous mes moments d'examens par leurs conseils et leur surprise sucrées, je leurs souhaite une vie pleine du bonheur et de succès.

A mes chères amies, qui ont tout que ce j'ai de plus chère au monde.

Fatma

Résumé

Le coronavirus est actuellement une maladie contagieuse majeure dans le monde. Le coronavirus 2019 (COVID-19) a été signalé pour la première fois à Wuhan et plus tard dans de nombreuses régions du monde. Cette maladie a été enregistrée presque dans le monde entier et provoque de graves effets secondaires.

L'industrie de la santé recherche avec impatience de nouvelles technologies intelligentes pour suivre et contrôler la croissance de l'épidémie de coronavirus dans cette crise sanitaire internationale. L'intelligence artificielle (IA) est l'une des plus grandes utilisations de la technologie mondiale à l'heure actuelle, qui permet de suivre et de détecter l'infection par la maladie Corona.

Dans cette note, nous proposons un modèle pour la classification de la maladie corona, qui a été développé en utilisant le modèle de réseau neuronal de contournement et basé sur un ensemble de données locales pour l'état de Tebessa. Le modèle atteint une précision de prédiction de 89% dans l'ensemble de données de test. Le processus de diagnostic de la maladie de Covid sera plus rapide et plus intelligent et aidera à alléger le fardeau du personnel médical.

Mots-clés : virus covid-19, apprentissage en profondeur, contournement du réseau neuronal, système de classification, Intelligence artificielle.

Abstract

Coronavirus is currently a major contagious disease worldwide. Coronavirus 2019 (COVID-19) was first reported in Wuhan and later in many parts around the world. This disease has been recorded almost worldwide and causes serious side effects.

The health industry is eagerly searching for new and smart technologies to track and control the growth of the coronavirus epidemic in this international health crisis. Artificial intelligence (AI) is one of the greatest uses of global technology at the moment, which can track and detect the infection with the Corona disease.

In this note, we propose a model for the classification of corona disease, which was developed using the bypass neural network model and based on a local data set for the state of Tebessa. The model achieves a prediction accuracy of 89% in the test data set. The process of diagnosing covid disease will be faster and smarter and help relieve the burden on the medical staff.

Keywords: covid-19, virus, deep learning, bypass neural network, classification system, Artificial intelligence.

الملخص

فيروس كورونا هو حاليًا مرض معدٍ كبير في جميع أنحاء العالم. تم الإبلاغ عن فيروس كورونا 2019 (COVID-19)، لأول مرة في ووهان ولاحقًا في أجزاء كثيرة في جميع أنحاء العالم. تم تسجيل هذا المرض تقريبًا في جميع أنحاء العالم وتسبب في آثار جانبية خطيرة.

تبحث صناعة الصحة بشغف عن تقنيات جديدة وذكية لتتبع نمو وباء الفيروس التاجي والسيطرة عليه في هذه الأزمة الصحية الدولية. يعد الذكاء الاصطناعي (AI) أحد أعظم استخدامات التكنولوجيا العالمية في الوقت الحالي، والذي يمكنه تتبع وكشف الإصابة بمرض الكورونا .

في هذه المذكرة نقترح نموذج تصنيف الإصابة بمرض الكورونا، تم تطويره باستخدام نموذج الشبكة العصبية الالتفافية وبالاعتماد على مجموعة بيانات محلية لولاية تبسة. يحقق النموذج دقة تنبؤ بنسبة % 89 في مجموعة بيانات الاختبار. ستكون عملية تشخيص الإصابة بمرض الكوفيد أسرع وأذكى وتساعد في التخفيف العبئ على الطاقم الطبي.

الكلمات المفتاحية: فيروس كوفيد-19، التعلم العميق، الشبكة العصبية الالتفافية، نظام تصنيف، الذكاء الاصطناعي.

Sommaires

| | |
|---------------------------|---|
| Introduction général..... | 1 |
|---------------------------|---|

Chapiter 1 : Deep Learning & Réseaux de neurones convolutifs

| | |
|---|----------|
| 1. Introduction | 3 |
| 2. Intelligence artificielle | 3 |
| 3. Machine Learning | 3 |
| 3.1. Apprentissage supervisé | 4 |
| 3.2. Apprentissage non supervisé | 4 |
| 4. Deep Learning..... | 4 |
| 4.1. Définition : | 4 |
| 4.2. Deep Learning VS Machine Learning..... | 5 |
| 4.2.1.Extraction des caractéristiques | 5 |
| 4.2.2.La performance..... | 5 |
| 4.3. Le fonctionnement..... | 5 |
| 4.4. Quelques algorithmes de Deep Learning | 7 |
| 4.4.1.Les réseaux de neurones profonds..... | 7 |
| 4.4.2.Les réseaux de neurones récurrents..... | 7 |
| 4.4.3.Les réseaux récurrent à mémoire court et long terme | 8 |
| 4.4.4.Les réseaux de neurones convolutifs..... | 8 |
| 5. Les réseaux de neurones Convolutifs..... | 8 |
| 5.1. L'opération de convolution | 9 |
| 5.2. Architecture de réseaux de neurone convolutif..... | 11 |
| 5.2.1.Couche convolution (CONV)..... | 11 |
| 5.2.2.Couche de pooling (POOL)..... | 11 |
| 5.2.3.Couche de correction (ReLU) | 12 |
| 5.2.4.Couche entièrement connectée (FC) | 12 |
| 5.2.5.Couche de perte (LOSS)..... | 13 |
| 5.3. Les paramètres de CNN | 13 |

| | |
|--|-----------|
| 5.3.1. Nombre de filtres..... | 13 |
| 5.3.2. Forme du filtre..... | 13 |
| 5.3.3. Forme du Max Pooling..... | 13 |
| 5.4. Les différents types de réseaux de neurones | 13 |
| 5.4.1. Adaline | 13 |
| 5.4.2. Les réseaux de RBF..... | 14 |
| 5.4.3. Réseaux de Hopfield | 14 |
| 5.4.4. Réseaux de Kohonen..... | 15 |
| 5.5. Exemple de modèles de CNN | 15 |
| 6. Conclusion | 15 |

Chapitre 2 : Les méthodes de détection des cas covid-19

| | |
|--|----|
| 1. Introduction | 17 |
| 2. Le Coronavirus | 17 |
| 3. La structure de coronavirus | 17 |
| 4. Symptômes et évolution | 18 |
| 5. Les types de coronavirus : | 19 |
| 6. Coronavirus en Algérie | 20 |
| 7. Les méthodes diagnostiques du covid-19..... | 21 |
| 7.1. Les tests d'amplification d'acide nucléique | 21 |
| 7.2. Les tests sérologiques :..... | 22 |
| 7.3. CT Scan | 23 |
| 8. Deep Learning dans le domaine médical | 23 |
| 9. Les méthodes de Détection d'objets sur des images | 24 |
| 9.1. La méthode de Viola et Jones..... | 24 |
| 9.2. La méthode de la transformée généralisée de Hough..... | 24 |
| 9.3. Méthode du réseau de neurone convolutif | 25 |
| 10. Deep learning pour identifier covid-19 | 25 |
| 10.1. Traitement | 26 |
| 10.1.1. ML et DL pour le développement de médicaments et de vaccins..... | 26 |
| 10.2. Diagnostique..... | 27 |
| 10.2.1. Approches basées sur l'analyse du son de la toux | 27 |
| 10.2.2. Approches basées sur des mesures non invasives | 27 |
| 10.2.3. Approches basées sur l'image médicale..... | 28 |
| 10.2.3.1. Approche basée sur l'image CT thoracique | 28 |

| | | |
|-----------|--|----|
| 10.2.3.2. | Approches basées sur l'image X-ray | 28 |
| 11. | Synthèse..... | 32 |
| 12. | Conclusion..... | 33 |

Chapitre 3 : Contribution et implimentation

| | | |
|--------|------------------------------------|----|
| 1. | Introduction | 34 |
| 2. | Problématique étudier..... | 34 |
| 3. | Conception globale..... | 34 |
| 3.1. | Collection de dataset | 35 |
| 3.2. | Prétraitement de dataset..... | 36 |
| 3.2.1. | Division de données | 37 |
| 3.2.2. | Augmentation de données | 37 |
| 3.3. | Création de model | 38 |
| 4. | Implémentation..... | 41 |
| 4.1. | Le langage de programmation | 41 |
| 4.2. | Le software | 42 |
| 4.3. | Le hardware | 45 |
| 5. | Réalisation | 46 |
| 5.1. | Connexion au google colab | 46 |
| 5.2. | Importer les bibliothèques | 47 |
| 5.3. | Diviser les données..... | 47 |
| 5.4. | Accès aux données | 48 |
| 5.5. | Data augmentation..... | 48 |
| 5.6. | Création de model | 49 |
| 5.7. | Génération de model | 50 |
| 5.8. | Compiler le model | 50 |
| 5.9. | Exécution le model..... | 51 |
| 6. | Résultats et discussion..... | 52 |
| 6.1. | Précision et erreur..... | 52 |
| 6.2. | Le rapport de classification | 53 |
| 6.3. | Matrice de confusion | 55 |
| 6.4. | Résultat de test..... | 56 |
| 7. | Conclusion..... | 57 |
| | Conclusion générale | 58 |

Liste des figures

| | |
|--|----|
| Figure 1:Topologie de réseau de neurones | 6 |
| Figure 2:Topologie de réseau de neurones profond | 7 |
| Figure 3:Architecture du CNN. | 9 |
| Figure 4:Schéma du parcours de la fenêtre de filtre sur l'image | 10 |
| Figure 5:Convolution avec une argument maximal. | 10 |
| Figure 6:Pooling avec un filtre 2x2 et un pas de 2 | 12 |
| Figure 7:La structure du coronavirus | 18 |
| Figure 8:Les symptômes de Coronavirus..... | 19 |
| Figure 9:carte de coronavirus en Algérie (20 Avril 2021) | 21 |
| Figure 10:Principales méthodes de diagnostic rapportées pour la détection du SRAS-CoV-2 | 22 |
| Figure 11:Exemple de Ct-Scan | 23 |
| Figure 12:Etat de l' Art les méthodes intelligentes pour diagnostiquer et traiter le covis-19... | 26 |
| Figure 13: Conception globale | 35 |
| Figure 14:Echantillant de dataset | 36 |
| Figure 15: Conception du model CNN. | 38 |
| Figure 16 : Architecture du model CNN proposé. | 40 |
| Figure 17: Connexion au Google Colab..... | 46 |
| Figure 18: Import des bibliographies | 47 |
| Figure 19:Division de données..... | 48 |
| Figure 20:Accès aux données..... | 48 |
| Figure 21:Augmentation de données..... | 49 |
| Figure 22:Architecture de model CNN | 50 |
| Figure 23:Génération de model..... | 50 |
| Figure 24:Compilation de model..... | 51 |
| Figure 25:Exécution de model | 51 |
| Figure 26 :Précision de model avec 50 époques | 52 |
| Figure 27:Erreur de model avec 50 époques | 53 |
| Figure 28:Rapport de classification (précision ,Recall, F1-score) | 55 |
| Figure 29:Matrice de confison | 56 |
| Figure 30:Resultat de test..... | 57 |

Liste des tableaux

| | |
|---------------------------------------|----|
| Tableau 1: Les travaux connexes | 29 |
|---------------------------------------|----|

Introduction

générale

1. Introduction

Le coronavirus est actuellement une maladie contagieuse majeure dans le monde. Le coronavirus 2019 (COVID-19), qui est causé par un virus appelé SARS-CoV-2, a été identifié pour la première fois à Wuhan, en Chine, en décembre 2019 et plus tard dans de nombreuses régions du monde ; Le 3 janvier 2020, l'Organisation mondiale de la santé a déclaré que le COVID-19 était une urgence de santé publique d'importance internationale et l'a confirmé comme une pandémie le 11 mars 2020. La maladie a été signalée dans 216 pays et territoires à travers le monde au 16 mai 2020. La maladie s'est propagée et a provoqué de graves effets secondaires. L'industrie de la santé recherche avec impatience de nouvelles technologies et technologies pour suivre et contrôler la croissance de l'épidémie de coronavirus dans cette crise sanitaire internationale. L'intelligence artificielle (IA) est l'une des plus grandes utilisations de la technologie mondiale à l'heure actuelle, qui permet de suivre la vitesse et la détection du taux de croissance du virus Corona, et d'identifier les risques et la gravité des patients atteints du virus Corona. L'IA peut également prédire la probabilité de décès grâce à une analyse appropriée des données antérieures sur les patients. L'intelligence artificielle peut nous aider à combattre le virus en testant des personnes, une assistance médicale, des données, des informations et des recommandations pour lutter contre la maladie. Pour résoudre des problèmes complexes dans nos vies, l'IA est un vaste parapluie qui se compose de nombreux sous-domaines. DL est un sous-ensemble de l'intelligence artificielle qui se concentre sur la création de modèles NN structurels profonds qui apprennent à partir de données à l'aide de plusieurs algorithmes. Les algorithmes DL incluent généralement des réseaux de neurones à convolution profonde (Deep CNN) qui, en analysant des images médicales, ces réseaux aident à diagnostiquer le COVID-19.

2. Objective de la thèse

Dans cette note, nous présentons la portée clé de l'IA axée sur la DL pour la recherche sur le COVID-19 qui comprend des aspects du diagnostic de la maladie. Notez qu'en raison de l'évolution rapide de la pandémie de COVID-19, de nombreux travaux de recherche publiés

ont déjà été cités avant une enquête approfondie, car ces travaux doivent déjà être analysés pour vérifier leur précision et leur qualité d'examen par les pairs. Le travail effectué dans cette mémoire consiste en :

- ✓ Un rapport sur les derniers systèmes de détection de la maladie Covid-19, basé sur des radiographies médicales.
- ✓ Proposition d'un système automatisé basé sur le contournement des réseaux de neurones pour diagnostiquer les personnes atteintes de la maladie Corona.
- ✓ Évaluer les performances du système proposé.

3. La structure de la thèse

1) Le premier chapitre : Deep Learning & Réseaux de neurones convolutifs

- ✓ Présente le contexte de l'intelligence artificielle , de Deep Learning et de Réseaux de neurones convolutifs

2) Le second chapitre : Les Méthodes de détection cas Covid-19

- ✓ Un état de l'art sur les travaux d'apprentissage automatique à la détection et la classification de covid-19.

3) Le troisième chapitre : Contribution & Implémentation

- ✓ Introduit notre contribution et l'implémentation de notre proposition, les résultats et la discussion

Chapitre 1

***Deep Learning & Réseaux de neurones
convolutifs***

1. Introduction

L'intelligence artificielle est une discipline scientifique qui vise à résoudre des problèmes d'une grande complexité logique. L'apprentissage automatique est l'un des domaines de l'intelligence artificielle qui permet à la machine d'apprendre et d'améliorer progressivement ses performances. Par conséquent, l'apprentissage en profondeur est un groupe de méthodes d'apprentissage automatique qui essayez de concevoir avec un haut niveau d'abstraction des données. Aujourd'hui, la technologie de l'apprentissage profond (Deep Learning) représente l'essentiel des recherches menées par les professionnels, d'autant plus qu'elle intervient dans plusieurs domaines comme la classification des images médicales. Dans ce chapitre, nous présenterons tout d'abord les concepts liés à l'apprentissage profond ainsi qu'un ensemble d'algorithmes de classification des images médicales.

2. Intelligence artificielle

L'intelligence artificielle (IA) vise à simuler la façon dont le cerveau humain prend des décisions. L'intelligence artificielle implique la mise en œuvre d'un certain nombre de technologies pour permettre aux machines de simuler une forme d'intelligence réelle. L'IA comprend divers sous-domaines, dont les plus connus sont l'apprentissage automatique et l'apprentissage profond.

3. Machine Learning

Selon [1] L'apprentissage automatique (machine Learning) est le champ de l'IA qui permet à la machine d'améliorer progressivement ses performances sur une tâche spécifique à partir des données sans données sans être explicitement programmé.

L'apprentissage automatique passe par les réseaux de neurones conçus pour imiter les capacités de prise de décision de l'homme, et est généralement divisé en :

3.1.Apprentissage supervisé

C'est la forme la plus commune d'apprentissage automatique, est une méthode permettant de transformer un jeu de données en un autre, le programme est formé sur un ensemble prédéfini d'exemples de formation, ce qui facilite ensuite sa capacité à parvenir à une conclusion précise lorsque de nouvelles données sont fournies [2] [3], tel que les algorithmes d'Arbres de décision, Régression logistique, et le plus connu c'est Machines à vecteurs de support (SVM).

3.2.Apprentissage non supervisé

Ce type partage une propriété commune avec l'apprentissage supervisé : il transforme un jeu de données en un autre. Mais l'ensemble de données dans lequel il se transforme n'est pas connu ou compris auparavant. Contrairement à l'apprentissage supervisé sera quant à lui alimenté uniquement par des exemples, et créera lui-même les classes qui lui semblent les plus judicieuses (clustering) ou des règles d'associations (algorithmes Apriori). L'algorithme K-moyen (K-means) permet de comprendre facilement le concept de classification non supervisée [4].

4. Deep Learning

4.1.Définition :

“ Le Deep Learning permet à des modèles composés de plusieurs couches de traitement d'apprendre des représentations des données avec de multiples niveaux d'abstraction ”. [5]

D'après les travaux de Deng L et al [5], le « Deep Learning » est une classe de techniques d'apprentissage automatique appartient au domaine de « Machine Learning » dans lesquelles de multiples couches de traitement de calcul itératif dans des architectures hiérarchisées supervisées sont exploitées pour les algorithmes d'apprentissage non supervisé pour des tâches d'analyse et de classification. L'apprentissage profond consiste essentiellement à calculer des caractéristiques hiérarchiques des paramètres des réseaux de neurones artificiels pour les représentations vectorielles des données d'observation ou d'entrées. La famille des méthodes

d'apprentissage en profondeur s'enrichit de plus en plus, englobant celles des réseaux de neurones, des modèles probabilistes hiérarchiques, ainsi que de nombreux algorithmes d'apprentissage des fonctionnalités supervisés et non supervisés.

4.2. Deep Learning VS Machine Learning

Deux caractéristiques principales distinguent Deep Learning de Machine Learning :

4.2.1. Extraction des caractéristiques

Dans la Machine Learning, Les fonctionnalités de l'application sont requises par un expert, puis codées manuellement, contrairement au Deep Learning dont les algorithmes tentent de connaître automatiquement les fonctionnalités de haut niveau des données.

4.2.2. La performance

Les performances sont associées à un volume de données accru. La précision des performances des algorithmes de Deep Learning s'améliore avec l'augmentation du volume de données l'inverse se produit avec les algorithmes de Machine Learning traditionnels où les résultats prévalent à mesure que la quantité de données augmente.

4.3. Le fonctionnement

Les méthodes d'apprentissage en profondeur dépendent principalement des réseaux de neurones, où le réseau de neurones se compose de trois couches importantes, comme le montre la figure :

-  La couche d'entrée ;
-  La couche cachée ;
-  La couche résultante.

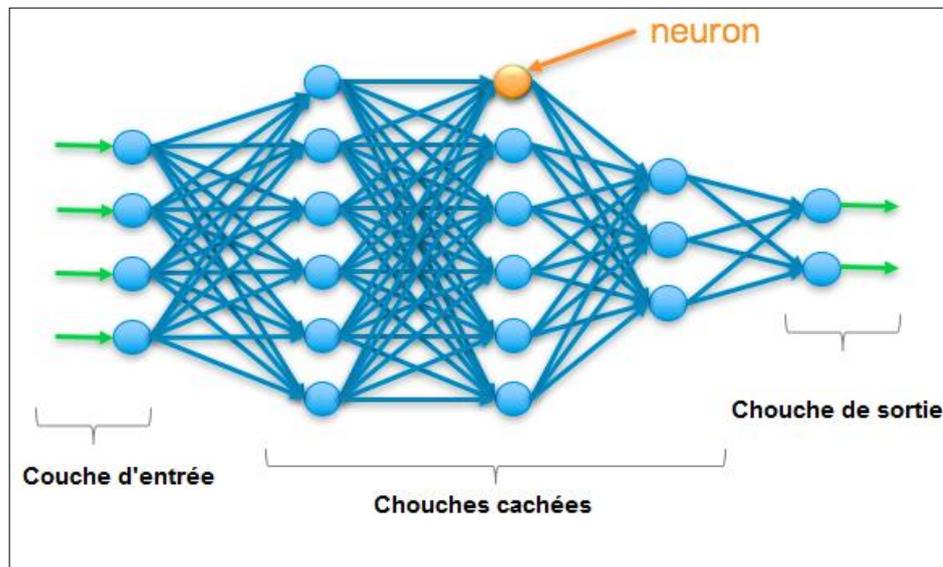


Figure 1: Topologie de réseau de neurones [6].

En deep learning, la profondeur du réseau est calculée en fonction du nombre de couches cachées dans le réseau neuronal, qui varie de 2 à 150 couches [6].

Le composant principal d'un réseau neuronal est constitué de ganglions (neurones) qui sont connectés les uns aux autres via des liens dirigés (flèche). Chaque flèche représente un lien entre une sortie de neurone et une autre entrée, et chaque flèche porte un poids (W). Chaque nœud est une unité de traitement qui reçoit des données sous forme de valeurs d'entrée et est passé à travers ce « réseau » de couches cachées jusqu'à ce qu'il converge dans la couche de sortie. La couche de sortie correspond à nos attentes : il peut s'agir d'un nœud (nous disons classification binaire) ou de quelques nœuds s'il s'agit d'un problème de classification multi-classes.

La forme à l'intérieur des neurones dans les couches de base représente une fonction d'activation qui peut être un Cube ,Elu , Hardsigmoid , Hardtanh , Identity , Leakyrelu ,Rationaltanh ,Relu ,RRelu ,Sigmoid , Softmax , Softplus ,Softsign , Tanh [7] .

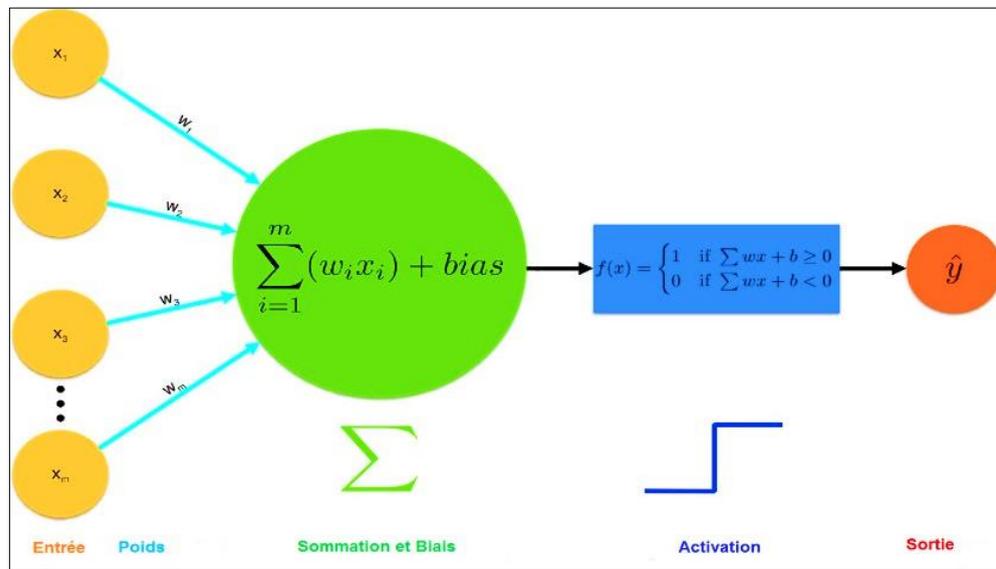


Figure 2: Topologie de réseau de neurones profond [7].

4.4. Quelques algorithmes de Deep Learning

Les architectures les plus utilisées en réseaux de neurones profonds sont :

4.4.1. Les réseaux de neurones profonds

Les réseaux de neurones profonds (Deep Neural Networks :DNN) sont des réseaux à flux direct dans lesquels les données circulent de la couche d'entrée à la couche de sortie sans revenir en arrière et les liens entre les couches sont à sens unique et ils ne touchent jamais un nœud de nouveau.

4.4.2. Les réseaux de neurones récurrents

Un réseau neuronal récurrents (Recurrent Neural Network :RNN) est une classe de réseaux neuronaux artificiels dans lesquels les connexions entre les nœuds forment un graphique vectoriel le long d'une séquence qui leur permet de montrer un comportement dynamique temporel d'une séquence temporelle. Les RNN peuvent utiliser leur état interne (mémoire) pour gérer la séquence d'entrée. Cela le rend applicable à des tâches telles que la

reconnaissance de l'écriture manuscrite ou la reconnaissance vocale.

4.4.3. Les réseaux récurrent à mémoire court et long terme

Les réseaux récurrents à mémoire court et long terme (Long short-term memory : LSTM) sont un type particulier de RNN, capable d'apprendre les dépendances à long terme qui permettent à RNN de se souvenir de choses qui se sont produites dans le passé et de trouver des modèles à travers le temps pour donner un sens à ses prochaines suppositions. Les LSTM ont battu des records pour l'amélioration de la traduction automatique, de la modélisation linguistique et du traitement linguistique multilingue.

4.4.4. Les réseaux de neurones convolutifs

Les réseaux de neurones convolutifs (Convolution Neural Network : CNN ou ConvNet) est une classe de réseaux neuronaux profonds qui est le plus couramment appliqué à l'analyse de l'imagerie visuelle. Leurs autres applications incluent la compréhension vidéo, la reconnaissance vocale et la compréhension du traitement du langage naturel.

5. Les réseaux de neurones Convolutifs

Le CNN est un type de réseau de neurones artificiels, il est largement utilisé dans le domaine de l'analyse d'images et la reconnaissance vidéo. CNN présente l'avantage d'utiliser des filtres de noyau (kernel) et mettent en œuvre du pooling.

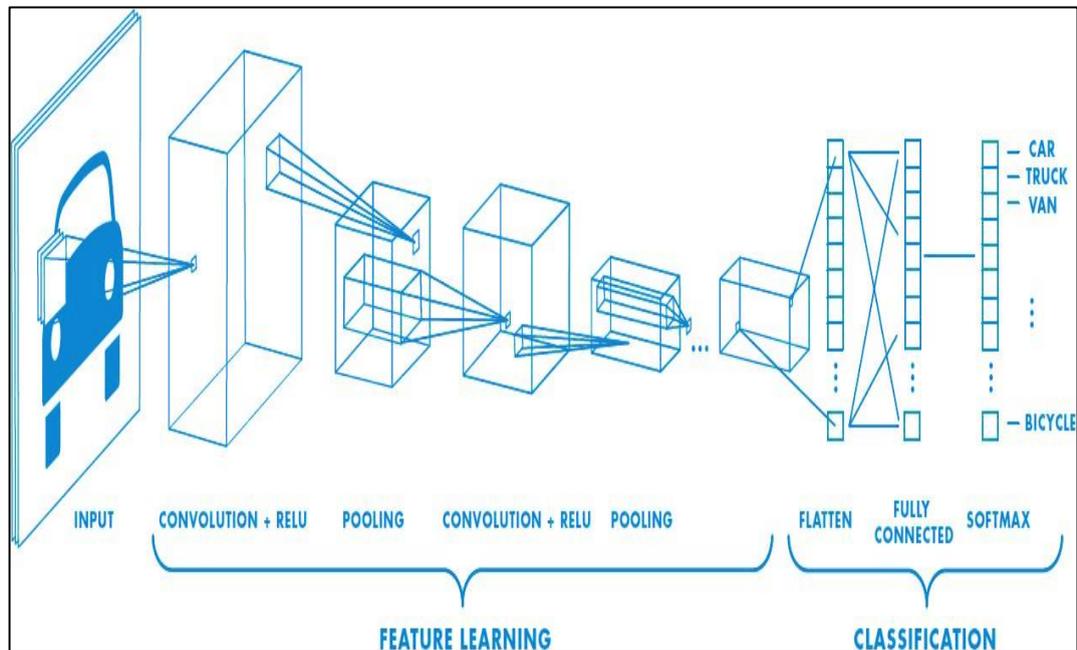


Figure 3: Architecture du CNN. [6]

5.1.L'opération de convolution

La convolution agit comme un filtrage. On définit une taille de fenêtre qui va se balader à travers toute l'image (rappelez-vous qu'une image peut être vue comme étant un tableau). Au tout début de la convolution, la fenêtre sera positionnée tout en haut à gauche de l'image puis elle va se décaler d'un certain nombre de cases (c'est ce que l'on appelle le pas) vers la droite et lorsqu'elle arrivera au bout de l'image, elle se décalera d'un pas vers le bas, ainsi de suite jusqu'à ce que le filtre ait parcouru la totalité de l'image [8] :

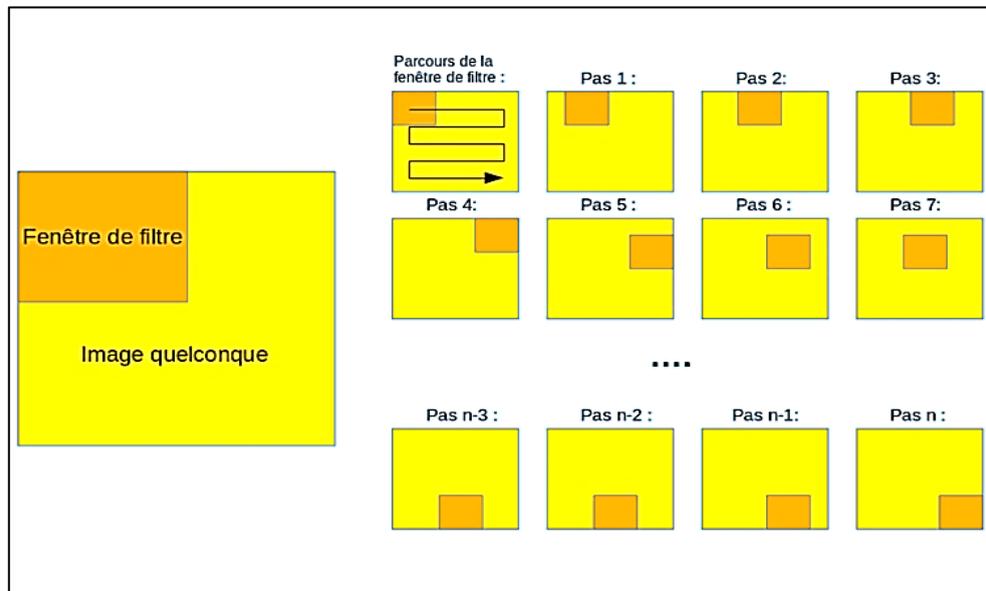


Figure 4:Schéma du parcours de la fenêtre de filtre sur l'image [8].

Le but est de se servir des valeurs présentes dans le filtre à chaque pas. Par exemple si l'on définit une fenêtre 3 par 3, cela représentera 9 cases du tableau (9 pixels). La convolution va effectuer une opération avec ces 9 pixels. Il peut s'agir de n'importe quelle opération, par exemple on extrait la valeur la plus grande (soit le pixel avec la plus grande valeur). [8]

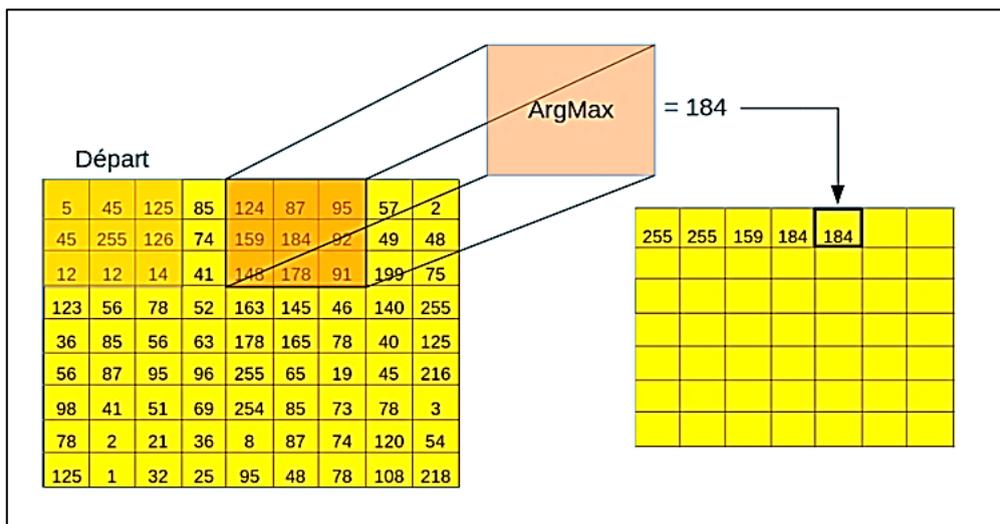


Figure 5:Convolution avec une argument maximal. [9]

On fait glisser la fenêtre en bleu et à chaque pas on récupère la valeur la plus grande parmi les 9 valeurs de pixels[8]. La sortie de cette opération (convolution) qui s'appelle « matrice de caractéristique » à des dimensions plus petites que celle de l'image en entrée

5.2. Architecture de réseaux de neurone convolutif

La structure de CNN consiste en un ensemble de couches de traitement indépendantes :

5.2.1. Couche convolution (CONV)

Cette couche est le composant principal des réseaux de neurones convolutifs et constitue leur première couche. Sa fonction est de localiser un ensemble de caractéristiques(features) dans les images qui ont été données sous forme des entrées, raison pour laquelle nous effectuons un processus appelé méta-filtrage, dans lequel nous faisons glisser la fonction d'image dans une forme de fenêtre et calculons le produit de convolution entre la feature et chaque portion de l'image balayée.

Pour mesurer le volume de la couche de convolution, nous avons besoin de 3 hyperparamètres :

- ✚ **Profondeur de la couche** : Le nombre de neurones qui ont une corrélation directe avec le même champ du récepteur.
- ✚ **Le pas** : Le pas contrôle le chevauchement des champs récepteurs. Plus le pas est petit, plus les champs récepteurs se chevauchent et plus le volume de sortie sera grand.
- ✚ **La marge (à 0) ou zéro padding** : parfois, il est commode de mettre des zéros à la frontière du volume d'entrée. Cette marge permet de contrôler la dimension spatiale du volume de sortie [10].

5.2.2. Couche de pooling (POOL)

La chose la plus commune à propos de cette couche est qu'elle est placée entre deux couches de convolution, ce qui lui permet de recevoir de nombreuses feature maps en entrée, l'opération de pooling consiste à minimiser la taille de l'image et à en préserver les propriétés importantes.

Parmi les caractéristiques connues de pooling est de donner une grande puissance de calcul, et

malgré la très faible représentation (et donc la perte d'informations associées), toutes les tendances actuelles consistent à utiliser des filtres de petite taille (type 2×2). Mais le risque de sur-apprentissage posait de nombreux problèmes qui permettaient très bien d'éviter la couche de pooling.

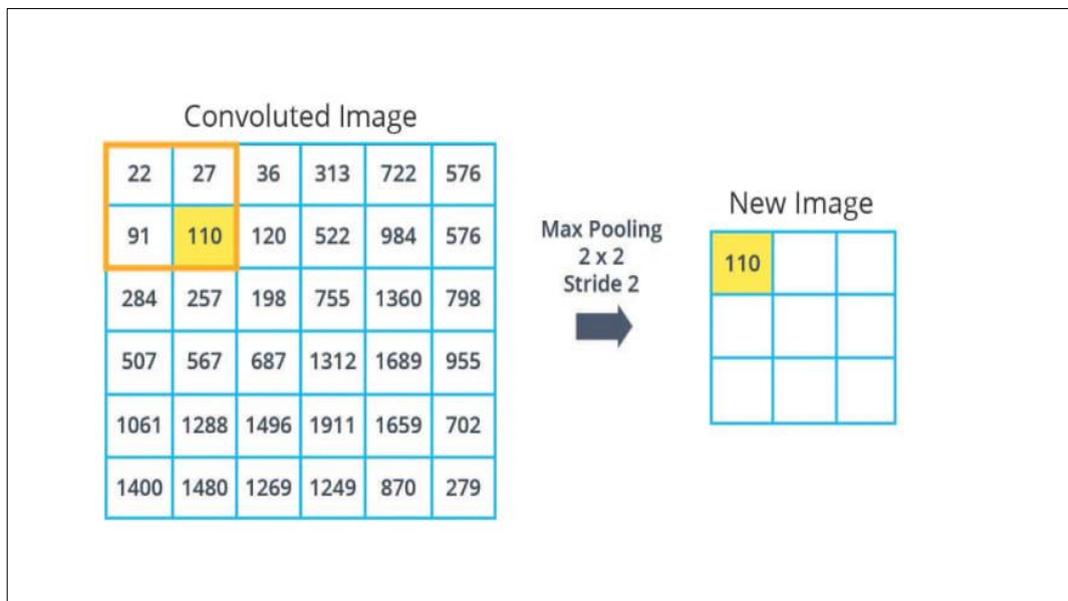


Figure 6: Pooling avec un filtre 2x2 et un pas de 2 [10].

5.2.3. Couche de correction (ReLU)

Les fonction d'activation tel que : Relu, tanh, sigmoid , on a but pour améliorer l'efficacité du traitement en. La fonction ReLU (Rectified Linear Units) désigne la fonction réelle non-linéaire, augmente les propriétés non linéaires de la fonction de décision elle remplace toutes les valeurs négatives reçues en entrées par des zéros.

5.2.4. Couche entièrement connectée (FC)

Ce type de couche reçoit un vecteur d'entrée et produit un nouveau vecteur de sortie. Pour cela, il applique une combinaison linéaire et éventuellement une fonction d'activation aux valeurs reçues en entrée.

5.2.5. Couche de perte (LOSS)

La dernière couche dans le réseau, spécifie comment l'entraînement du réseau pénalise l'écart entre le signal prévu et réel. Diverses fonctions de perte adaptées à différentes tâches peuvent y être utilisées. La perte « Softmax » est utilisée pour prédire une seule classe parmi K classes mutuellement exclusives [10].

5.3. Les paramètres de CNN

5.3.1. Nombre de filtres

En raison de la réduction de la taille des images intermédiaires en fonction de la profondeur du traitement, le nombre de filtres dans les calques proches de l'entrée est réduit. Il est très important de conserver le nombre de sorties intermédiaires (c'est-à-dire de multiplier le nombre d'images intermédiaires par le nombre de positions de pixels) afin de préserver les informations d'entrée.

5.3.2. Forme du filtre

Les formes de filtrage sont souvent choisies en fonction de l'ensemble de données, en raison de la grande diversité de leurs formes dans la littérature. Les meilleurs résultats sont ceux obtenus sur MNIST (28×28) dans une plage de 5×5 dans la première couche.

5.3.3. Forme du Max Pooling

Les valeurs typiques sont 2×2 . Des volumes d'entrée très importants peuvent justifier une mise en commun de 4×4 dans les couches inférieures. Cependant, le choix de formes plus grandes réduira considérablement la dimension du signal et peut entraîner une perte excessive d'informations.

5.4. Les différents types de réseaux de neurones

5.4.1. Adaline

Adaline (L'ADAPtive LINear Element) est un réseau neuronal simple-couche, avec plusieurs

nœuds, chacun acceptant plusieurs entrées et générant une sortie. L'adaline possède des valeurs d'activations continues, une fonction d'activation linéaire est basé sur la règle d'apprentissage de WIDROW & HOFF¹, en effet cette loi minimise l'erreur quadrique, en utilisant la méthode du gradient. Il opère sur les exemples bien classés et mal classés en optimisant toujours les coefficients des hyperplans séparateurs [11].

5.4.2. Les réseaux de RBF

Les réseaux de neurones à base radiale sont une classe particulière des réseaux de neurones multicouches. L'idée générale des réseaux RBF dérive de la théorie d'approximation des fonctions. La raison de son application vient du fait que le réseau utilise des fonctions gaussiennes standard qui sont à symétrie radiale. Son apprentissage est basé sur l'algorithme K-means et l'algorithme des moindres carrées [12].

Un réseau de neurones RBF se compose de trois couches à savoir : une couche d'entrée, une couche cachée et une couche de rendement. Chaque couche se compose d'un certain nombre de neurones, la couche d'entrée passe seulement les données d'entrée à la couche cachée qui contient les k noeuds, qui appliquent une transformation non linéaire sur les variables d'entrée.

Les RBFs sont principalement utilisés pour résoudre des problèmes d'approximation de fonctions dans des espaces de grandes dimensions.

5.4.3. Réseaux de hopfield

Hopfield Neural Network est un modèle réseau de neurones récurrents discret dans le temps, dans lequel un neurone est mis à jour par unité de temps. Ce réseau est constitué de N neurones avec des états binaires (-1, 1 ou 0, 1) et est basé sur la loi de Hebb². Ainsi, le poids de la connexion entre deux cellules nerveuses augmente lorsque les deux neurones sont activés en même temps.

¹ Le perceptron de Frank Rosenblatt est très proche de la règle de Hebb, la grande différence étant qu'il tient compte de l'erreur observée en sortie.

² Donald Hebb a établi la règle Hebb en 1949. Elle est utilisée comme hypothèse en neurosciences car cette règle indique que lorsque deux neurones sont excités ensemble, un lien est créé ou renforcé entre eux. Cette théorie tente d'expliquer l'apprentissage associatif, dans lequel une association est formée en répétant deux stimuli. La répétition d'un seul stimulus provoque le rappel de l'autre.

5.4.4. Réseaux de kohonen

Ce réseau dépend de la recherche d'un modèle neuronal plus proche de la réalité, car il simule des observations biologiques du fonctionnement du système nerveux de perception chez les mammifères. La loi modifiée (autoriser l'oubli) de Hebb est utilisée pour l'apprentissage. La communication est améliorée dans l'état où le neurone connecté a une activité synchrone et diminue lorsque le contraire se produit. Il est couramment utilisé dans les applications de classification et de traitement d'image.

5.5.Exemple de modèles de CNN

Dans les réseaux neuronaux, un réseau neuronal convolutif est largement utilisé pour la reconnaissance d'images, les classifications d'images, la détection d'objets et la reconnaissance faciale. CNN a apporté des changements révolutionnaires à l'industrie médicale où l'imagerie est la source de données médicales en croissance rapide sous la forme de rayons X, de tomodensitomètres, de diagnostics de la tuberculose, de pré-diagnostic par rayons X, d'IRM, de mammographie, etc.

En plus de détecter des anomalies dans les radiographies pulmonaires. L'apprentissage en profondeur peut être utilisé efficacement pour détecter des anomalies dans les rayons X afin de détecter diverses maladies à l'aide d'images open source.

6. Conclusion

Dans ce chapitre, nous explorons la technologie d'apprentissage en profondeur et tentons de clarifier les concepts qui nous intéressent dans ce domaine, y compris la classification et les implications pour la relation entre l'apprentissage automatique et l'apprentissage en profondeur. Nous avons mentionné comment cette technologie a surmonté les inconvénients des méthodes traditionnelles en rendant les systèmes moins complexes et plus rapides.

Nous ne nous sommes pas arrêtés à ce stade car nous avons besoin de réseaux appelés réseaux de neurones (CNN), spécialisés dans la classification des images. Ces réseaux sont capables

d'extraire des caractéristiques des images présentées en entrée et de classer ces propriétés. Cependant, les réseaux de neurones convolutifs présentent un certain nombre de limitations, d'une part, les hyperparamètres du réseau sont difficiles à pré-évaluer. En fait, le nombre de couches ou le nombre de neurones par couche ou encore les différentes connexions entre les couches sont des éléments cruciaux et sont déterminés principalement par une bonne intuition ou une série de tests / calcul d'erreurs (ce qui est coûteux en temps). Dans le chapitre suivant, nous parlerons du concept de méthodes de détection des maladies à l'aide d'images et prendrons la maladie de Covid 19 comme étude de cas.

Chapitre 2

Les Méthodes de détection cas

Covid-19

1. Introduction

L'émergence de la pandémie de coronavirus a suscité l'intérêt pour les modèles prédictifs capables de prédire la propagation du virus, notamment pour améliorer et soutenir les processus de prise de décision. Dans ce chapitre, nous fournissons d'abord un aperçu complet du coronavirus, puis discutons des principales approches d'apprentissage en profondeur qui visent à prédire la propagation de la maladie dans l'espace et le temps. L'objectif est de montrer les tendances émergentes dans ce domaine de recherche et de fournir une perspective générale sur les moyens possibles de traiter ce problème. Enfin, nous aborderons les principaux avantages et inconvénients des différents modèles, et soulignerons les tendances de développement prometteuses pour améliorer ces approches.

2. Le Coronavirus

Les coronavirus forment une famille de virus variés (Coronaviridae) qui peuvent infecter aussi bien l'homme que l'animal. Leur nom signifie "virus en couronne" et vient du fait qu'ils possèdent tous un aspect en forme de couronne lorsqu'ils sont observés au microscope. Les coronavirus ont été identifiés pour la première fois chez l'humain dans les années 1960. Il s'agit de virus causant des maladies émergentes, c'est-à-dire des infections nouvelles dues à des modifications ou à des mutations du virus. Les coronavirus humains causent principalement des infections respiratoires, allant du rhume sans gravité à des pneumopathies sévères parfois létales. Ils peuvent aussi s'accompagner de troubles digestifs tels que des gastro-entérites.

3. La structure de coronavirus

La figure 7 présente la structure de coronavirus, L'enveloppe est formée des protéines S (spike), M et M'(membranaires) et E (enveloppe). La nucléocapside (NC), formée par l'ARN

génomique associé à la protéine N, est contenue dans la capside, elle-même entourée de l'enveloppe.

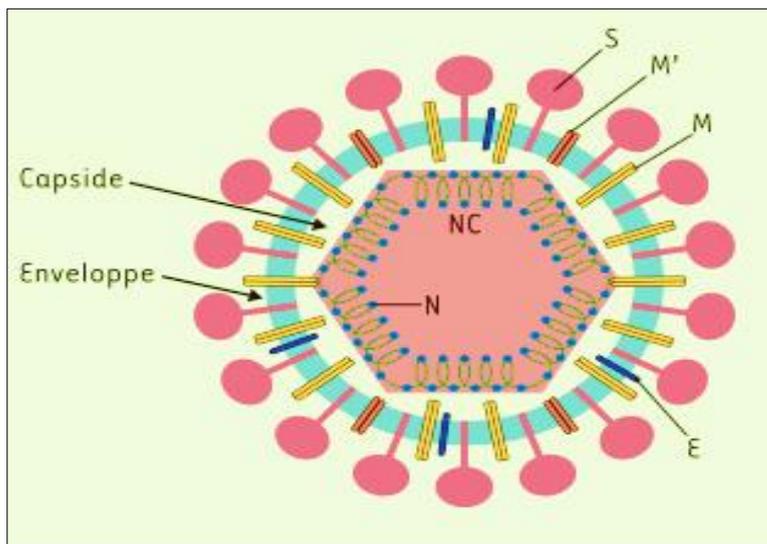


Figure 7: La structure du coronavirus

4. Symptômes et évolution

Les manifestations des coronavirus font leur apparition moins de 24 heures après l'infection. Le plus généralement, le virus entraîne des **maladies respiratoires légères à modérées** comme le rhume avec des symptômes tels que :

- ✚ Mal de tête,
- ✚ Toux,
- ✚ Gorge irritée,
- ✚ Fièvre,
- ✚ Un sentiment général de malaise

Dans le cas du covid-19, une perte de goût et d'odorat font partie des nouveaux symptômes évoqués.

Plus gravement, il peut provoquer des maladies respiratoires des voies inférieures comme la **pneumonie** ou la **bronchite**, particulièrement chez les personnes atteintes d'une maladie cardio-pulmonaire, chez celles dont le système immunitaire est affaibli et les personnes âgées.

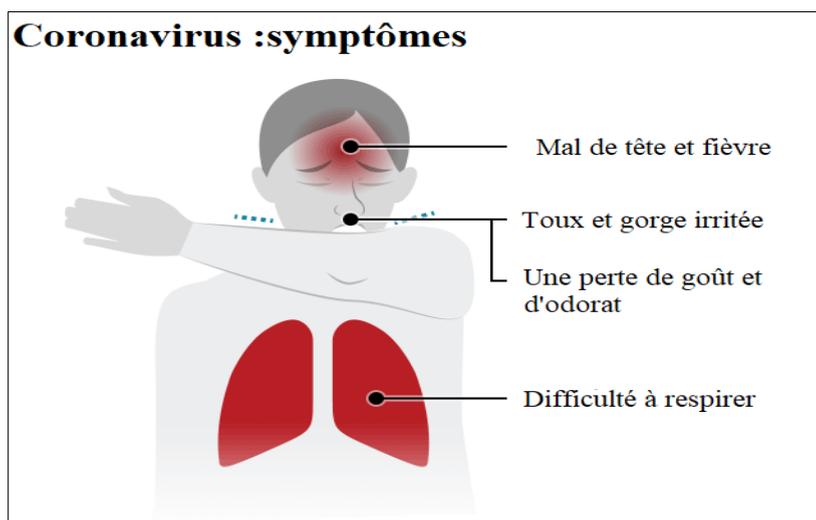


Figure 8: Les symptômes de Coronavirus.

Le **temps d'incubation** correspond à la durée entre l'exposition au virus et à la manifestation des premiers symptômes. Pour les coronavirus, il est évalué **entre 3 et 5 jours** dans la majorité des cas, mais peut durer dans des cas extrêmes de 12 et 14 jours.

5. Les types de coronavirus :

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) :

1. 229E (alpha coronavirus)
2. NL63 (alpha coronavirus)
3. OC43 (bêta-coronavirus)
4. HKU1 (bêta-coronavirus)

Et trois plus graves :

5. SRAS-CoV (le bêta-coronavirus qui cause le Syndrome respiratoire aigu sévère, ou SRAS, identifié en Chine en 2002).
6. MERS-CoV (le coronavirus bêta qui provoque le syndrome respiratoire du Moyen-Orient, ou MERS découvert en 2012 en Arabie Saoudite).
7. Sars-CoV-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.

6. Coronavirus en Algérie

La pandémie de Covid-19 est une crise sanitaire majeure provoquée par une maladie infectieuse émergente apparue fin 2019 en Chine. Ce virus est à l'origine d'une pandémie mondiale, 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.

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.

Des cas de Covid-19 sont ensuite détectés, il y avait 3158 décès et 119805 cas confirmés en Algérie au 20 avril 2021. Parmi eux, 1085 cas dans la wilaya de Tébessa (Figure 9).

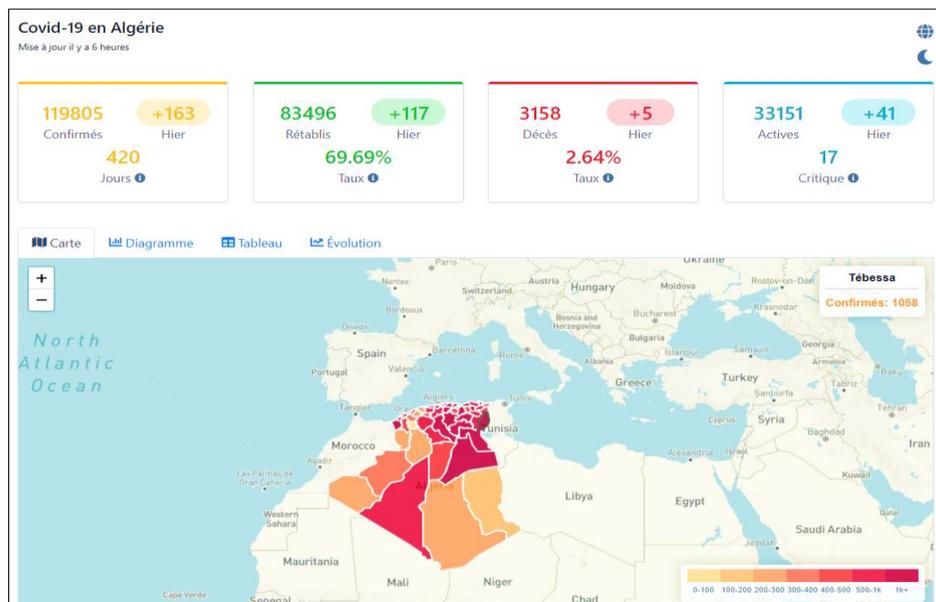


Figure 9 : carte de coronavirus en Algérie (20 Avril 2021) [13]

7. Les méthodes diagnostiques du covid-19

7.1. Les tests d'amplification d'acide nucléique

Les tests RT-PCR sont les tests de référence pour le diagnostic de la phase aiguë du Covid-19. Fondés sur des méthodes de biologie moléculaire, ils permettent de détecter la présence du SARS-CoV-2 dans l'organisme d'un individu à un instant T et donc de confirmer un diagnostic de Covid-19 posé par un médecin. Concrètement, des échantillons nasopharyngés sont recueillis chez les personnes qui présentent des symptômes de la maladie par le biais d'un écouvillon introduit dans la narine jusqu'au rhinopharynx. À l'heure actuelle, la période idéale pour détecter l'ARN viral est de 1 à 7 jours après l'apparition des symptômes. Au-delà, le prélèvement nasopharyngé n'est plus optimal, comme l'indique la Haute Autorité de santé (HAS) dans un récent rapport. D'autres prélèvements (salivaires par exemple) peuvent alors être envisagés pour établir un diagnostic.

7.2. Les tests sérologiques :

Sont faits à partir d'un échantillon de sang. Mais les modalités de réalisation peuvent varier. On distingue deux catégories principales :

Les tests ELISA automatisables et les tests unitaires. Les premiers ne peuvent être faits qu'au sein d'un laboratoire de biologie médicale, compte-tenu du plateau technique nécessaire à leur analyse. Ils sont réalisés sur prélèvement sanguin, généralement par prise de sang, et apportent des informations sur la présence de différents types d'anticorps. Ces tests entrent dans la catégorie des examens de biologie médicale ;

Les tests unitaires sont quant à eux réalisés à partir d'une goutte de sang prélevée au bout du doigt, en laboratoire, mais aussi en pharmacie, en cabinet médical ou chez soi.

Avec ces tests, il ne s'agit pas d'un dosage, mais de la détection d'une quantité donnée d'anticorps. La réponse est alors binaire : si le sang contient moins d'anticorps que le seuil minimal, le résultat est négatif ; s'il en contient plus, le résultat est positif.

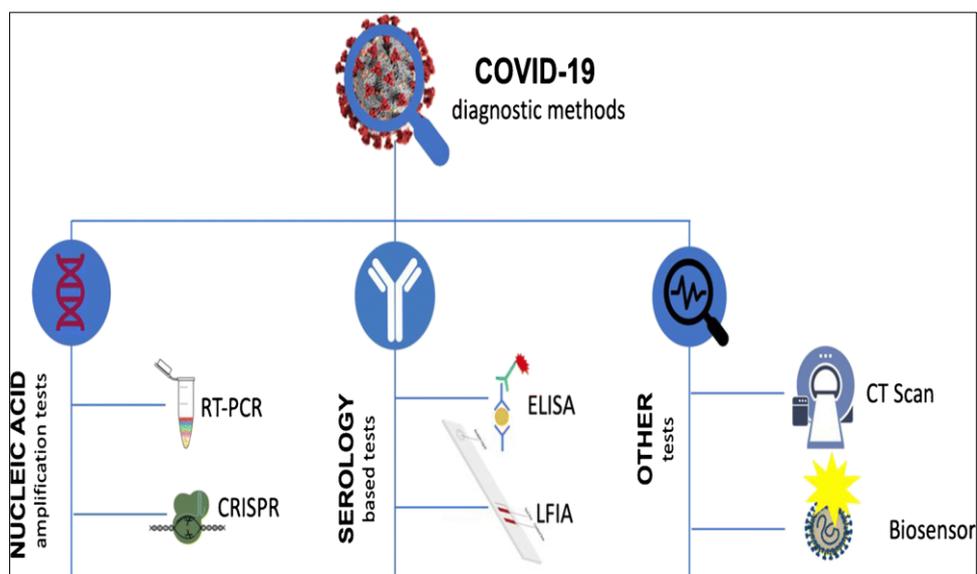


Figure 10: Principales méthodes de diagnostic rapportées pour la détection du SRAS-CoV-2 [14]

7.3.CT Scan

Le meilleur mode diagnostique c'est le scanner thoracique. La tomodensitométrie (TDM) thoracique (ct scan) apporte le meilleur diagnostic pour la pneumonie COVID-19. L'analyse des données d'imagerie et de tests de laboratoire de plus de 1.000 patients montre que la TDM thoracique surpasse les tests biologiques dans le diagnostic de l'épidémie associée au nouveau coronavirus, COVID-19. Les chercheurs concluent que la tomodensitométrie devrait être utilisée, dès que possible, comme principal outil de dépistage.

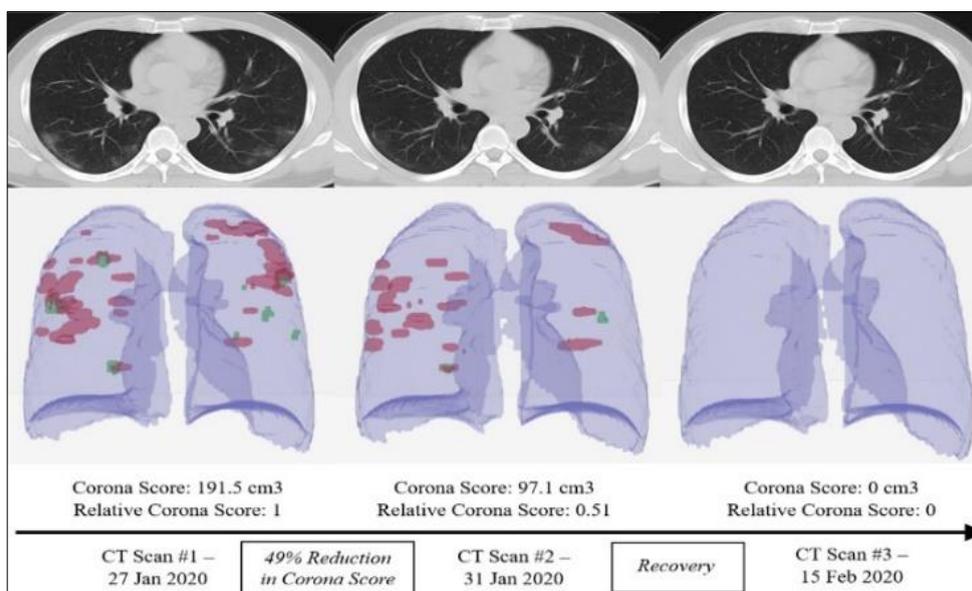


Figure 11 : Exemple de Ct-Scan [15]

8. Deep Learning dans le domaine médical

L'utilisation efficace de l'apprentissage en profondeur dans le secteur médical a la capacité de développer un nouvel outil qui surpasse les techniques actuelles en termes de précision de classification et de complexité temporelle. La classification des images était l'un des principaux domaines dans lesquels les réseaux de neurones ont apporté une contribution importante à l'analyse d'images médicales. L'analyse faite par [16] donne un aperçu des

différentes techniques de traitement d'image appliquées aux images médicales.

Cette partie résume les modèles et les techniques de prétraitement utilisés dans la classification d'images covid-19.

9. Les méthodes de Détection d'objets sur des images

La détection d'objet dans une image fait partie des tâches très simples pour un humain mais qui sont très compliquées à mettre en place pour un ordinateur. Dans cette partie nous groupons les méthodes de détection d'objet sur les images en 3 méthodes :

9.1. La méthode de Viola et Jones

Le principe de cette méthode repose sur des calculs de caractéristiques entre deux zones constituées d'un ou plusieurs rectangles chacune. Chaque rectangle est représenté par la somme des pixels qu'il contient, la différence des deux zones donne une idée du contraste entre deux zones, on obtient alors une caractéristique.

Cependant la méthode de Viola et Jones ne permet pas réellement de détecter un objet, elle détecte plutôt un motif qui correspond aux caractéristiques qu'elle a retenu lors de son entraînement, si bien qu'un même objet sous un angle différent peut s'avérer totalement étranger à cette méthode de reconnaissance.

9.2. La méthode de la transformée généralisée de Hough

La transformée de Hough est une méthode de détection de forme. Sa première forme avait pour but de détecter et décrire les droites présentes dans une image. Elle a ensuite été étendue pour détecter des formes plus complexes (cercles et ellipses), et une transformée de Hough dite généralisée permet de détecter des formes quelconques.

9.3. Méthode du réseau de neurone convolutif

Le Réseau de neurone convolutif (RNC) est presque devenu une norme dans la détection d'objet par ordinateur dans le domaine de reconnaissance d'image. L'idée de cet algorithme revient souvent en apprentissage automatique. On met en place une formule qui donne une valeur de l'erreur, et on dérive cette formule par rapport à chaque poids dans le système, ainsi si la dérivée est négative c'est qu'il diminue l'erreur alors on peut itérativement augmenter ce poids, autrement on le diminue (là aussi itérativement).

Le RNC est très fiable et sophistiqué mais il est très coûteux en calcul et n'est pas capable de localiser un objet dans l'image. Depuis il a été amélioré en RCNN (pour Region-based Convolutional Neural Network) qui va d'abord détecter les zones d'intérêt (même texture, couleur etc...) et les découper pour les donner à un RNC, ainsi on a la nature de l'objet et sa place dans l'image.

10. Deep learning pour identifier covid-19

L'industrie de la santé est à la recherche de nouvelles technologies et techniques pour suivre et contrôler la croissance de l'épidémie de coronavirus dans cette crise sanitaire internationale. L'une des plus grandes utilisations de la technologie mondiale à l'heure actuelle est l'intelligence artificielle (ML et DL), qui peut :

- ✓ Suivre la vitesse et détecter le taux de croissance du virus corona,
- ✓ Identifier le risque et la gravité des patients atteints du virus Corona,
- ✓ Anticiper la possibilité de décès en analysant de manière adéquate les données antérieures des patients.
- ✓ Aider à lutter contre le virus en testant des individus, une assistance médicale, des données et des informations, et des recommandations concernant le contrôle des maladies.

Dans cette partie, nous présentons le champ principal de l'IA en se concentrant sur le ML et le DL vers la recherche sur le COVID-19 qui intègre les aspects du diagnostic de la maladie et des développements de médicaments et de vaccins. La figure 12 montre une taxonomie d'enquête sur les travaux de recherche sur la ML et la DL pour le diagnostic et le traitement du COVID-19.

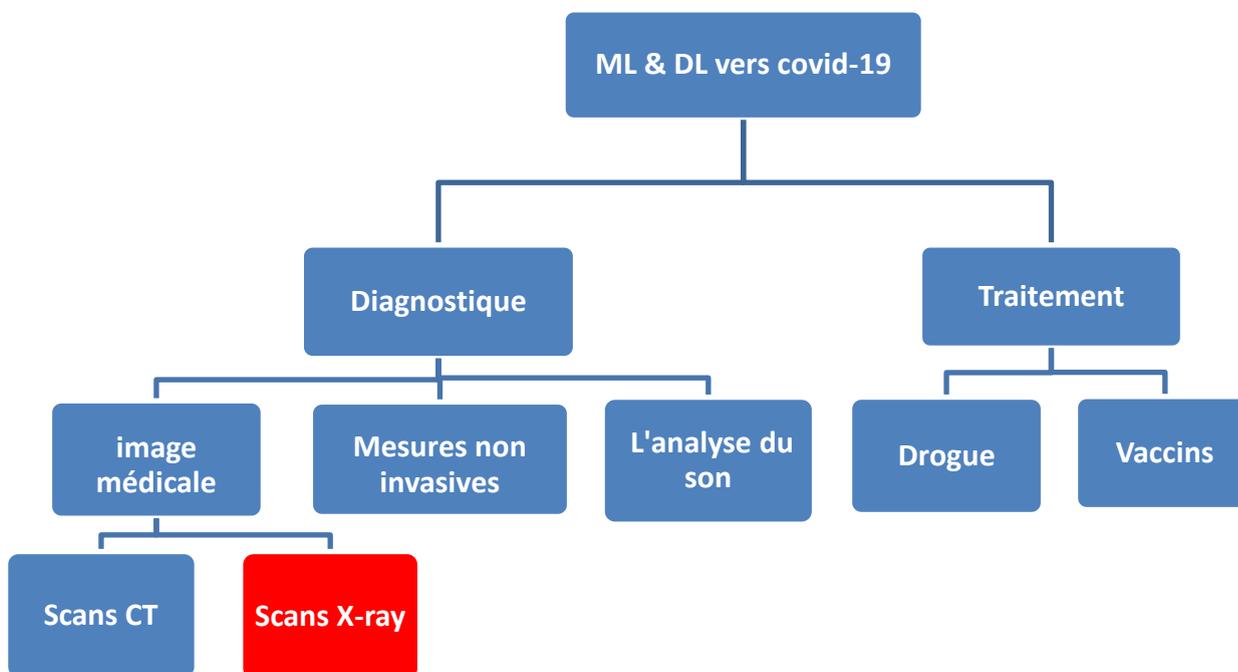


Figure 12 : Etat de l'Art sur les méthodes intelligentes pour diagnostiquer et traiter le covid-19.

10.1. Traitement

10.1.1. ML et DL pour le développement de médicaments et de vaccins

Divers programmes de développement de médicaments et vaccins contre le SRAS-CoV-2 et le COVID-19 ont été suggérés pour se concentrer sur les études de protéomique et de génomique. L'une des principales contributions à la médecine intelligente est l'utilisation de la

ML et de la DL dans le développement de nouveaux médicaments et vaccins.

10.2. Diagnostique

10.2.1. Approches base sur l'analyse du son de la toux

Des recherches récentes ont commencé à étudier comment les sons respiratoires (par exemple, la toux, la respiration et la voix) dans un hôpital par les appareils de patients testés positifs pour le COVID-19 diffèrent de ceux des personnes en bonne santé.

Dans une recherche [17], une application téléphonique a été conçue pour collecter des données sonores et rapporter une analyse pour identifier le COVID-19 sur la base de la toux collectée, en utilisant une combinaison de 48 COVID-19, 102 bronchites, 131 coqueluches et 76 sons de toux normaux pour apprendre et évaluer ce diagnostic méthode. Où les équations mathématiques sont utilisées pour convertir les données regroupées sur la toux en une échelle Mel m pour le prétraitement des données.

Critique :

La toux est un symptôme de plus de 30 conditions médicales autres que le COVID-19. Cela rend l'identification de l'infection au COVID-19 par la toux seule un grand défi pour divers problèmes, car les médecins ont recours dans la plupart des cas à l'utilisation de signaux sonores émis par le corps humain, comme les gémissements, la respiration, les battements du cœur, la digestion et les vibrations.

10.2.2. Approches basées sur des mesures non invasives

Maghdid et coll. [18] a prévu le système de diagnostic invisible COVID-19 basé sur des capteurs de téléphone intelligent, dans le but de rassembler les caractéristiques possibles de la maladie du patient. Par exemple, grâce à une fonction d'enregistrement, les capteurs peuvent détecter la voix d'un patient et détecter la température corporelle d'un patient via une fonction de reconnaissance des doigts. Les données collectées sont ensuite transférées vers un serveur

cloud financé par l'IA pour le diagnostic et l'analyse des maladies.

Critique :

Il faut un certain temps pour comparer différentes images tomodensitométriques, et les radiologues ne peuvent pas terminer physiquement leur compréhension.

10.2.3. Approches basées sur d'image médicale

Les travaille dans cette partie son classe en deux grand catégories, les approches basé sur la tomographie assistée par ordinateur (CT : computerized tomography) et les approche base sur Scans aux rayons X (X-ray : X-radiation) .

10.2.3.1. Approche base sur l'image CT thoracique

Même si l'interprétation manuelle d'un scanner peut prendre jusqu'à 15 minutes, l'IA peut analyser les images en 10 s. Par conséquent, le traitement d'image avancé avec un réseau neuronal artificiel a la possibilité d'améliorer considérablement la fonction de la tomodensitométrie dans la détection du COVID-19 en permettant à une grande proportion de patients d'identifier la maladie facilement et rapidement avec précision. La poursuite des tests d'imagerie CT basés sur l'IA implique généralement les étapes suivantes : division régionale de la région d'intérêt (ROI), élimination du tissu pulmonaire, identification de l'infection régionale et classification du COVID-19.

10.2.3.2. Approches basées sur l'image X-ray

Les scans thoraciques ont été proposées comme une méthode très utile pour évaluer et tester les patients atteints de COVID-19. Par rapport aux images tomodensitométriques, les images radiographiques thoraciques (CXR) sont plus simples à acquérir dans les examens de radiologie clinique.

Il existe plusieurs méthodologies d'apprentissage en profondeur telles que CNN qui sont utilisées pour trouver une détection meilleure et plus rapide dans la détection du COVID-19

sur les images radiographiques Dans les centres médicaux et les hôpitaux, les appareils à rayons X offrent des résultats moins coûteux et plus rapides grâce à la numérisation différente. organes humains. L'interprétation de nombreuses images radiographiques est généralement effectuée manuellement par les radiologues. Les modèles pré-entraînés ont rendu la détection du COVID-19 beaucoup plus facile et plus rapide. D'une autre coté, il existe des travaux base sur l'image x ray utilisent les algorithmes de deep learning pour classifier la gravite de covid-19.

Le tableau suivant résume quelques travaux de recherche utilisent l'image x ray pour détecter le covid-19 avec différents datasets :

Tableau 1: Les travaux connexes

| Travaux | Data set | Deep Learning méthode | Le rôle de méthode | Taux | Sensitivité | Auc |
|-----------------------------|---|-----------------------|--|---------|-------------|-----|
| Narin et al. [19] | Seulement 50 scans aux rayons X COVID-19 et 50 pour les non-COVID-19. | Pre-trained ResNet-50 | Classification binaire : Être porteur ou non de COVID. | 98% | N/A | N/A |
| Wang and Wong [20] | 16756 scans aux rayons X de 13645 patients atteints de COVID-19 | CNN | | 92.4% | 91.0% | N/A |
| Maghdid et al. [21] | Un total de 170 COVID-19. | CNN | Classification binaire : Être porteur ou non de COVID. | 98% | 100% | N/A |
| Apostolopoulos and Mpesiana | Un total de 1428 scans aux rayons X (224 COVID-19, | Pre-trained VGG-19 | Classification binaire : Être porteur ou non de | 93.48 % | 92.85% | N/A |

| | | | | | | |
|----------------------------|--|---|--|---------|--------|-----|
| [22] | 700 patients atteints de pneumonie et 504 normaux). | | COVID. | | | |
| Sethy and Behera [23] | Scans aux rayons X de 25 COVID-19 positifs et 25 négatifs. | ResNet-50 et SVM | Classification binaire : Être porteur ou non de COVID. | 93.28 % | 97% | N/A |
| Hemdan et al. [24] | Scans aux rayons X de 25 COVID-19 positive et 25 patients normales . | CNN | Classification binaire : Être porteur ou non de COVID. | 90% | N/A | N/A |
| Hall et al. [25] | Un total de 455 scans aux rayons X (135 de COVID-19 et 320 de pneumonie virale et bactérienne. | Pre-trained ResNet-50 | Classification binaire : Être porteur de virus COVID ou virus de la pneumonie | 89.2% | 95% | N/A |
| Apostolopoulos et al. [26] | Un total de 3905 scans aux rayons X pour plusieurs classes de maladies, y compris le COVID-19. | MobileNet-v2 | Classification multi class : Covid ou autres maladies | 99.18 % | 97.36% | N/A |
| Loey et al. [27] | 307 scans aux rayons X pour le virus COVID-19, normal, de la pneumonie et des bactéries . | GAN with AlexNet, GoogLeNet, and ResNet18 pre-trained models. | Classification binaire : Être porteur de virus COVID ou virus de la pneumonie. | 80.56 % | N/A | N/A |

| | | | | | | |
|------------------|--|---|--|---------|-----|-----|
| Maia et al. [28] | Total de 437 images radiographiques (217 COVID-19, 108 autres maladies et 112 saines). | Different Convolutional Support Vector Machines (CSVMs) | Classification ternaire : Maladie de COVID, maladie de seine ou autre maladie. | 98.14 % | N/A | N/A |
|------------------|--|---|--|---------|-----|-----|

Critique

La détection COVID-19 basée sur des images est actuellement confrontée à plusieurs obstacles, tels que :

- Rareté et indisponibilité des données d'entraînement à grande échelle : de nombreuses techniques de DL basées sur l'IA dépendent de données d'entraînement à grande échelle, y compris l'imagerie médicale. Cependant, les ensembles de données disponibles pour l'IA sont insuffisants, en raison de l'explosion rapide du COVID-19. En réalité, l'interprétation des échantillons de formation prend du temps et peut nécessiter des professionnels de la santé qualifiés.
- Données bruyantes et rumeurs en ligne. Les défis émergent du fait de dépendre des médias sociaux portables en ligne ; sans aucun changement significatif, de grandes informations d'image et de faux rapports sur COVID-19 ont été signalés dans différents points de vente en ligne. Cependant, pour juger et filtrer les données d'image et d'erreur, les algorithmes ML et DL basés sur l'IA semblent être lents.
- Confidentialité et protection des données : le coût de la collecte de données personnelles à l'ère du big data et de l'IA est très faible.

Tanque tout les critiques de cette partie son concerner le problème de data set, alors nous avons choisi de collecter une data set spécifier de notre région pour que notre travaille soit spécifier aux COVID-19-Tebessa.

11. Synthèse

L'épidémie de COVID-19 a eu un effet profond sur le bien-être des personnes dans le monde entier et le nombre de décès liés à la maladie continue de croître dans le monde. Bien que la technologie ait pénétré notre vie quotidienne avec un grand succès, en particulier en ML et DL, l'IA a également contribué à soutenir les gens dans la difficile bataille contre le COVID-19. DL n'est qu'un moyen efficace de fournir des solutions prometteuses axées sur les données pour aider l'humanité à gérer le COVID-19.

Dans ce chapitre, nous avons exploré les méthodes ML et DL basées sur l'IA pour le diagnostic et le traitement du COVID-19.

Nous avons d'abord étudié les deux approches les plus importantes spécialisées dans tous les cas de maladie Corona, à savoir l'approche thérapeutique et l'approche diagnostique. De plus, nous nous sommes plongés dans l'approche diagnostique, où nous avons découvert trois sous-approches spécialisées dans le diagnostic par le son, par rayons X ou par plusieurs autres facteurs incontrôlés. Nous avons identifié les faiblesses de chaque approche, en concluant que le diagnostic par rayons X est l'approche la plus précise pour diagnostiquer la maladie corona. À ce titre, nous avons étudié 10 travaux connexes (pour les années 2020 et 2021) spécialisés dans le diagnostic par rayons X, car chaque étude repose sur un algorithme d'apprentissage en profondeur différent ainsi que sur un ensemble de données spécial.

La maladie Corona étant une maladie du début de l'année 2020, ce virus est en constante évolution et prend plusieurs formes qui diffèrent d'un pays à l'autre, aussi notre étude a également montré que le point faible de ces 10 travaux, c'est le manque de données et le recours à la collecte manuelle qui nécessite une coopération avec un groupe médical professionnel. À partir de là, nous devions d'abord collecter un ensemble de données sur la maladie Corona, le classer manuellement, puis choisir un algorithme d'apprentissage pour atteindre des résultats pouvant être comparés au reste des 10 résultats des travaux cités précédents.

Notre travail est similaire à celui de détection de la maladie covid-19 basé sur les images X ray à la différence près que nous avons formé le model avec une dataset spécifique à notre

région « Willaya de Tébessa » pour éviter les critiques citer précédents et aussi d'augmenter le taux de modèle.

De plus, cette étude base également à l'algorithme de CNN dans le but d'établir une comparaison entre notre taux est les taux de recherche spécifier à la classification des image x ray.

12. Conclusion

Dans ce chapitre, nous avons présenté en premier point le covid 19, y compris la structure, les symptômes, les types et leur émergence dans notre région, en plus nous avons étudié les techniques traditionnelles de détection de covid 19. Pour rendre les systèmes plus rapides et plus faciles nous avons établi un état de l'art sur les méthodes intelligentes de traitement et diagnostique de covid 19, finalement une synthèse sur les travaux base sur les images x-ray est présentée à la fin de ce chapitre.

Le chapitre qui suit, on va présenter notre contribution ainsi que notre modèle les algorithmes utilisés dans notre architecture.

Chapitre 3

Contribution & Implémentation

1. Introduction

Dans ce chapitre, nous présentons notre Système de détection de la maladie de covid 19 a la base d'image X-ray. En premier temps, nous débutons par une introduction qui constitue une brève définition de notre système de classification automatique, ensuite nous présenterons notre contribution de classification proposées en détaillerons les différents modules qui le composent.

2. Problématique étudier

Les modèles actuels pour déterminer la détection du COVID-19 sur la base de l'image radiographique sont d'excellents modèles qui donnent de bons résultats et présentent de grands avantages pour le personnel médical et le diagnostic des maladies. Avec la situation actuelle que le monde connaît depuis un an et la diversité des types d'espèces de coronavirus, il est difficile de trouver un ensemble de données fiables pour la détection de la maladie Corona 19. Seul un petit nombre de données ont été trouvées, dont la plupart étaient privées et inaccessibles. Dans cette optique, et puisque notre état (Willaya de Tébessa) dispose d'un hôpital public en charge des maladies à coronavirus (Organisation hospitalière publique de Bougra Bouarras, Bakaria), nous l'avons renforcé pour créer un ensemble de données pour notre état dans le but de développer un système automatisé capable de déterminer la découverte de Corona 19 en tant que système spécifique à l'état Tébessa.

3. Conception globale

Dans cette section, nous allons essayer de donner une vue générale de notre système.

Dans notre système, nous traiterons les données "image de type X-ray" qui seront divisées en deux parties (données d'entraînement et données de test) et passeront par la phase de prétraitement(*) , comme illustré à la figure suivante. Ensuite, nous créons un modèle

d'apprentissage qui prend les entrées de données traitées à l'étape (*) et s'entraîne sur une partie des données (données d'entraînement). Le système final consistera à insérer une nouvelle image X-ray et à classer en covid positive / négative.

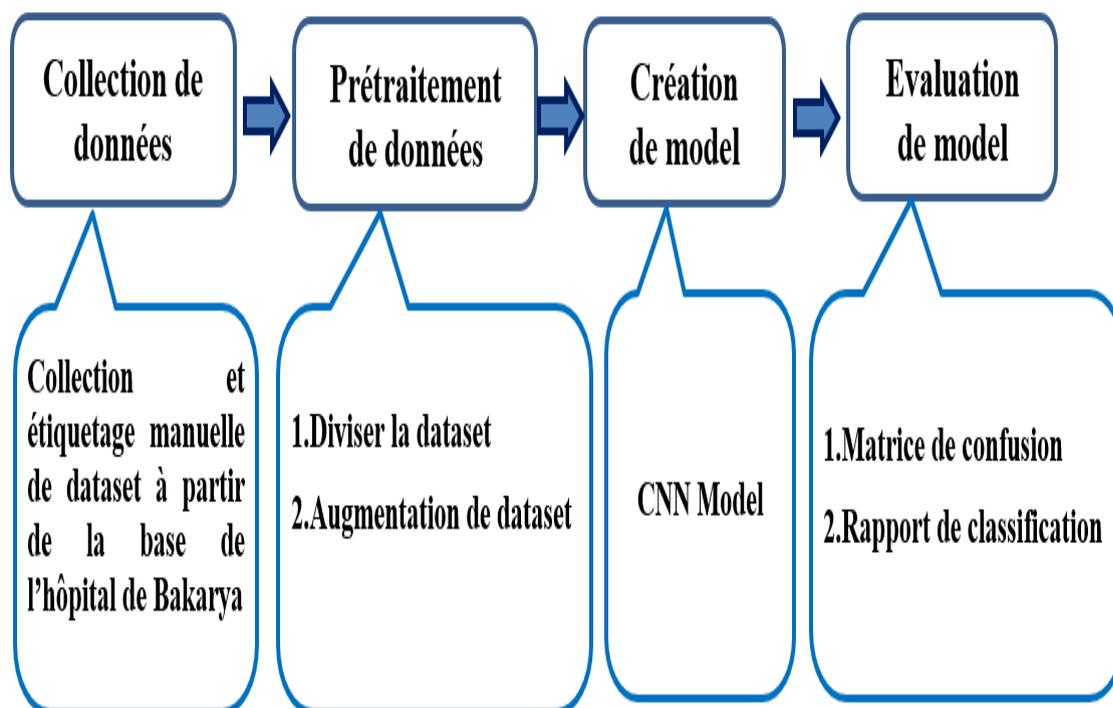


Figure 13: Conception globale

3.1. Collection de dataset

L'hôpital Bougher Bouras fournit un ensemble de données de tomodensitométrie de personnes porteuses de COVID-19 et de personnes en bonne santé, et il est donc varié.

L'ensemble de données est confronté au problème qu'il n'est pas toxique, nous avons donc utilisé l'expertise d'un médecin pour aider à étiqueter les données en positives (portez la maladie de covid) et négatives (ne portez pas la maladie de covid).

L'ensemble de données contient 139 images sur 512 x 512 enregistrées au format JPG.

Les images ont été classées en deux catégories :

- ✓ 69 pour la catégorie des maladies à covid positif.
- ✓ 70 pour la catégorie des maladies à covid négatif.

Des exemples de ces photos sont présentés dans la figure suivante :

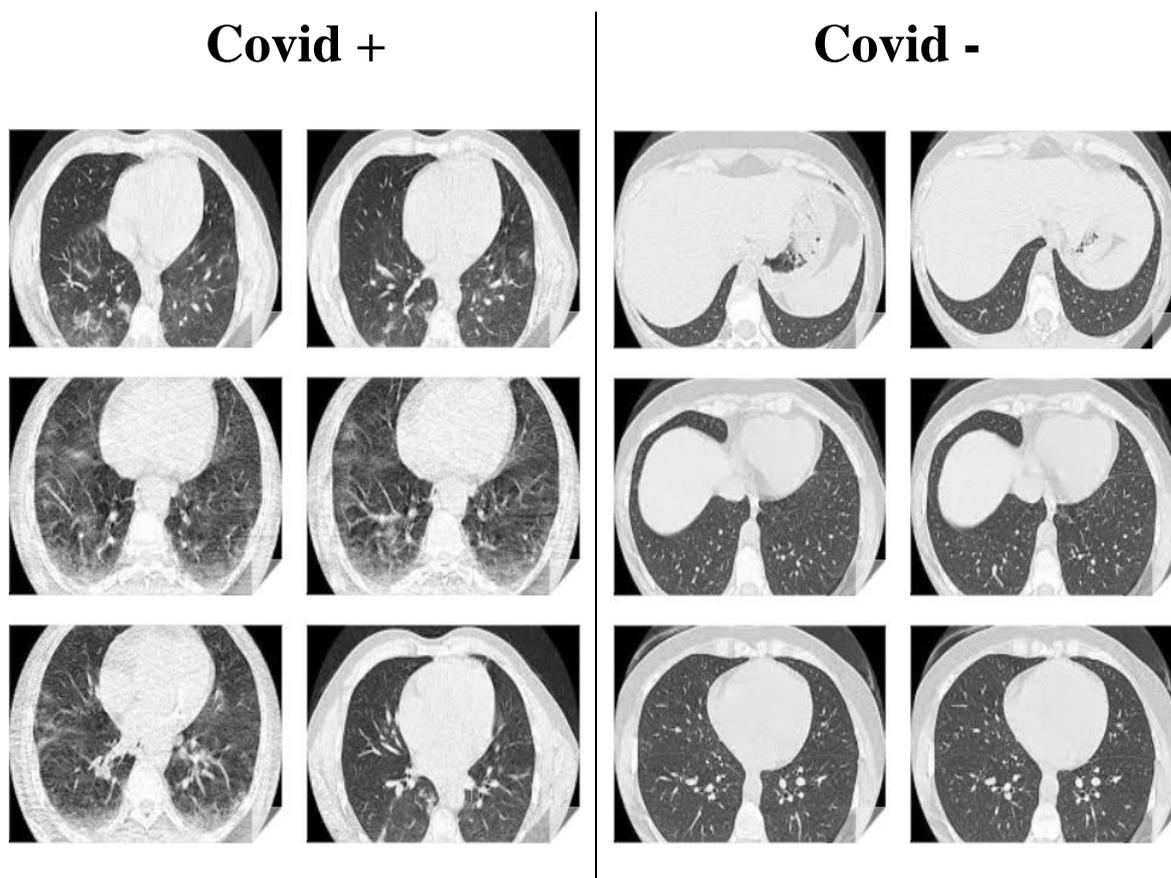


Figure 14: Echantillon de dataset

3.2. Prétraitement de dataset

Après avoir sélectionné l'ensemble de données, une étape très importante dans toute analyse d'image est le traitement d'image. Les images doivent être traitées et modifiées de manière à être pleinement compatibles pour obtenir une plus grande efficacité du modèle conçu. Pour

alimenter les images d'entrée dans le formulaire, les images sont normalisées pour une meilleure lisibilité. Les images ont été reconfigurées selon les exigences de la forme construite. L'augmentation des données est effectuée sur l'ensemble de données d'entraînement pour éviter le surajustement.

3.2.1. Division de données

Pour augmenter les performances du modèle, les images d'entrée ont été prétraitées à l'aide de l'augmentation des données a été effectuée sur l'ensemble de données.

L'ensemble de données a été divisé en un rapport 80:20. L'ensemble de données d'apprentissage contient 111 images et le test contient 28 images.

3.2.2. Augmentation de données

L'augmentation des données résout les problèmes de sur-apprentissage (Overfitting) en augmentant le volume de données. Ceci est fait sur la base de la conviction que des informations supplémentaires peuvent être obtenues grâce à des renforts à partir de l'image d'entraînement.

Il existe plusieurs techniques d'augmentation des données disponibles, mais il est très nécessaire de conserver la photo de l'affiche tout en appliquant un zoom.

Les réseaux de neurones convolutifs ont reçu beaucoup d'attention dans la classification des maladies. Il est très important que le modèle s'entraîne avec le Big Data. Lorsqu'un réseau de neurones est formé sur de petites données, il y a un risque de sur-apprentissage. Le modèle est trop lorsqu'il fonctionne bien sur les données d'entraînement, mais il est faible sur tout nouvel ensemble de données. Diverses approches ont été recommandées au cours de la période pour réduire le surajustement. Dans ce scénario, l'ensemble de données d'entraînement a été augmenté à l'aide des techniques suivantes :

- ✓ Retournement d'image

- ✓ Zoomer sur les images

Ces techniques ont été appliquées à la fois aux ensembles de traitement et de test.

3.3.Création de model

Le modèle que nous présentons dans la figure 2 est composé de trois bloc (Bloc A, Bloc B et Bloc C). Le Bloc A est composé de deux couches de convolution et une couche de Maxpooling avec un dropout de 25%, le bloc B et C sont les deux composés d'une seule couche de convolution et une seule couche avec Maxpolling avec un dropout de 25%.

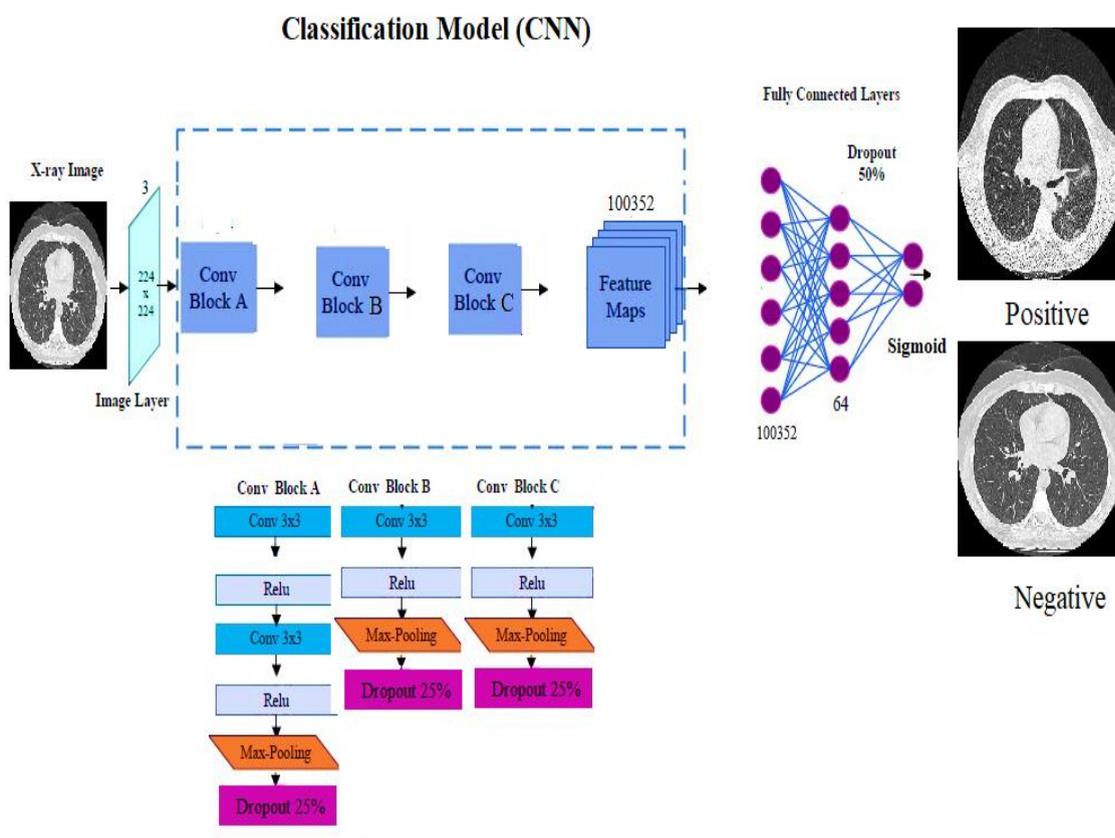


Figure 15: Conception du model CNN.

L'image en entrée est de taille $224*224*3$, l'image passe d'abord à la première couche de convolution. Cette couche est composée de 32 filtres de taille $3*3$, Chacune de nos couches de convolution est suivie d'une fonction d'activation ReLU cette fonction force les neurones à retourner des valeurs positives, après cette convolution 32 feature maps de taille $32*32$ seront créés.

Les 32 feature maps qui sont obtenus auparavant ils sont donnés en entrée de la deuxième couche de convolution qui est composée de 64 filtres, une fonction d'activation RELU est appliquée sur la couche de convolution, ensuite on applique Maxpooling de taille $2*2$ avec un pas de 2 pour réduire la taille de l'image ainsi la quantité de paramètres et de calcul. À la sortie de cette couche, nous aurons 64 feature maps de taille $112*112$, la fin de ce bloc on applique un dropout de 25%.

Les sorties de ce bloc sont données en entrée du bloc 2, une couche de convolution composée de 64 filtres de taille $3*3$ suivie d'une fonction d'activation ReLU est appliquée, ensuite une couche maxpooling de taille $2*2$ avec un pas de 2 est aussi appliquée et nous aurons 64 feature maps de taille $56*56$, la fin de ce bloc aussi un dropout de 25% est appliqué.

La même chose pour le Bloc c, mais cette fois on applique 128 filtres de taille $3*3$, et on aura finalement 128 feature maps de taille $28*28$.

Le vecteur de caractéristiques issu des convolutions a une dimension de 100352 ($28*28*128$). Après ces trois blocs, nous utilisons un réseau de neurones composé de trois couches fully connected. La première couche est de 100352 neurones, la deuxième est composée de 64 neurones et le dernier est composé de 2 neurones avec la fonction d'activation Sigmoid qui permet de calculer la distribution de probabilité des 2 classes (nombre de classes dans notre Dataset).

Les paramètres de chaque couche décrits dans cette partie sont bien détaillés dans la figure 16.

Model: "Binary_classification"

| Layer (type) | Output Shape | Param # |
|------------------------|-----------------------|---------|
| img_input (InputLayer) | [(None, 224, 224, 3)] | 0 |
| layer_1 (Conv2D) | (None, 224, 224, 32) | 896 |
| layer_2 (Conv2D) | (None, 224, 224, 64) | 18496 |
| layer_3 (MaxPooling2D) | (None, 112, 112, 64) | 0 |
| dropout (Dropout) | (None, 112, 112, 64) | 0 |
| layer_4 (Conv2D) | (None, 112, 112, 64) | 36928 |
| layer_5 (MaxPooling2D) | (None, 56, 56, 64) | 0 |
| dropout_1 (Dropout) | (None, 56, 56, 64) | 0 |
| layer_6 (Conv2D) | (None, 56, 56, 128) | 73856 |
| layer_7 (MaxPooling2D) | (None, 28, 28, 128) | 0 |
| dropout_2 (Dropout) | (None, 28, 28, 128) | 0 |
| fc_1 (Flatten) | (None, 100352) | 0 |
| lyaer_8 (Dense) | (None, 64) | 6422592 |
| dropout_3 (Dropout) | (None, 64) | 0 |
| predictions (Dense) | (None, 2) | 130 |

=====
 Total params: 6,552,898
 Trainable params: 6,552,898
 Non-trainable params: 0

Figure 16 : Architecture du model CNN proposé.

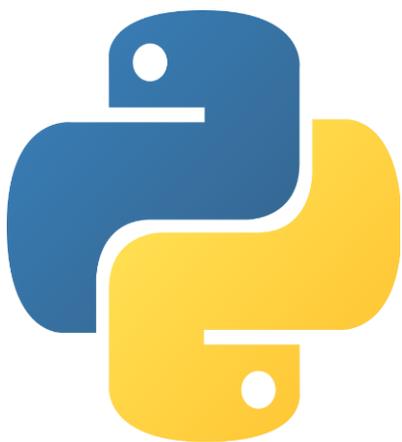
4. Implémentation

Dans cette partie nous présenter dans une première partie le langage de programmation python utilisé dans l'implémentation de model et les outils software tel que les bibliographies importer :

Keras , Tensorflow , Matplotlib , Numpy , Sklearn , ImageDataGenerator ...

Et le hardware utilise pour implémenter notre modèle, ensuite nous exprimant les résultats obtenus.

4.1.Le langage de programmation



Python : Version 3.7.10

Est un langage de programmation interprété, multi-paradigme et multiplateformes. Il favorise la programmation impérative structurée, fonctionnelle et orientée objet. Il est doté d'un typage dynamique fort, d'une gestion automatique de la mémoire par ramasse-miettes et d'un système de gestion d'exceptions.

Il est placé sous une licence libre proche de la licence BSD4 et fonctionne sur la plupart des plates-formes informatiques. Il est conçu pour optimiser la productivité des programmeurs en offrant des outils de haut niveau et une syntaxe simple à utiliser.

4.2. Le software



Keras : Version 2.5.0

Est considéré comme une puissante de la bibliothèque Python, facile à utiliser pour développer et évaluer des modèles d'apprentissage en profondeur.

Il a un design minimaliste qui permet de construire un réseau couche par couche ; l'entraîner et l'exécuter.

Il englobe les bibliothèques de calcul numérique efficaces Theano et TensorFlow et permet de définir et de former des modèles de réseaux neuronaux en quelques courtes lignes de code.

Il s'agit d'une API (Application programming interface) de réseau neuronal de haut niveau, aidant à utiliser largement l'apprentissage profond et l'intelligence artificielle. Il s'exécute au-dessus d'un certain nombre de bibliothèques de niveau inférieur, notamment TensorFlow, Theano, etc.

Le code Keras est portable; nous pouvons implémenter un réseau de neurones dans Keras en utilisant Theano ou TensorFlow comme back-end sans aucun changement de code.



Tensorflow : Version 2.5.0

Est une bibliothèque de logiciels open source pour le calcul numérique de haute performance. Créée par l'équipe de Google dans le de faciliter la création des modules d'apprentissage automatique. Son architecture flexible permet un déploiement facile du calcul sur diverses plates-formes (CPU, GPU, TPU), des ordinateurs de bureau aux clusters de serveurs. Même si TensorFlow a été conçu pour les réseaux de neurones, il fonctionne bien pour d'autres réseaux où le calcul peut être modélisé sous forme de graphique de flux de données. TensorFlow utilise également plusieurs fonctionnalités de Theano telles que l'élimination commune et sous-expression, la différenciation automatique, les variables partagées et symboliques.



Sklearn : Version 0.22.2. post1

Scikit-learn est une bibliothèque Python, libre et dédiée à l'apprentissage automatique. Elle comprend des fonctions pour estimer des régressions logistiques, des algorithmes de classification, et les machines à vecteurs de support. Elle est conçue pour s'harmoniser avec les autres bibliothèques libre Python, notamment NumPy et SciPy

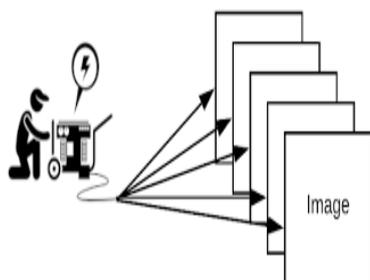


Image data generator

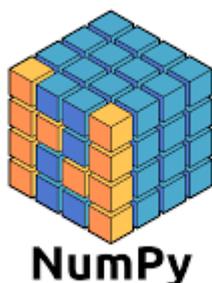
Cette fonction génère des lots de données d'image de tenseurs avec une augmentation des données en temps réel. Les données seront bouclées (par lots). Keras possède cette classe ImageDataGenerator qui permet aux utilisateurs d'agrandir l'image à la volée très facilement.

Utilisé si votre statut d'utilisation n'est pas très courant et que vous ne disposez que de quelques photos dans votre jeu de données d'entraînement



Matplotlib : Version 3.2.2

Est une bibliothèque de traçage pour le langage de programmation Python et son extension mathématique numérique NumPy C'est un logiciel de premier ordre qui fait de Python un concurrent averti à des outils scientifiques tels que MatLab ou Mathematica. elle conçue pour la génération de visualisations simples et puissantes . La bibliothèque est prise en charge par différentes plates-formes et utilise différents kits d'interface graphique pour la représentation des visualisations résultantes. Les différents IDE (comme IPython) prennent en charge la fonctionnalité de Matplotlib.

Numpy : Version 1.19.5

Numpy est une bibliothèque pour effectuer des opérations arithmétiques numériques en Python. Le package de base autour duquel se construit la pile de calcul scientifique est appelé numpy (Numerical PYthon). La bibliothèque Numpy fournit une gestion plus facile des tables de nombres et des fonctions complexes (propagation). Elle fournit également une abondance de fonctionnalités utiles pour les opérations sur les n-tableaux et les tableaux en python, en plus des conseils fournis pour les opérations mathématiques de type de tableau numpy qui améliore les performances et accélère ainsi l'exécution.

4.3.Le hardware

Le Deep Learning est un domaine avec des exigences en calculs intenses et la disponibilité des ressources (surtout en GPU) dédiés à cette tâche vont fondamentalement influencer sur l'expérience de l'utilisateur car sans ses ressources, il faudra trop de temps pour apprendre de ses erreurs ce qui peut être décourageant. alors nous avons choisi la plateformes Google Colaboratory pour pour effectuée les expérimentations .



Google Colab : ou Colaboratory est un service cloud, offert par Google (gratuit), basé sur Jupyter Notebook et destiné à la formation et à la recherche dans l'apprentissage automatique. Cette plateforme permet d'entraîner des modèles de Machine Learning directement dans le cloud.

5. Réalisation

5.1. Connexion au google colab

Premièrement nous Téléchargeons l'ensemble de données téléchargé sur Google Drive à partir du compte Gmail. Après cela, montez le lecteur google sur le bloc-notes colab comme illustré à la figure 17. Nous Cliquons sur l'URL et sélectionnez le compte Gmail et entrons le code d'authentification.

```
# connecter au google colab
from google.colab import drive
drive.mount('/content/drive')
```

Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).

Figure 17: Connexion au Google Colab

5.2.Importer les bibliothèques

Pour commençais le prétraitement, et l'augmentation de donnés, la figure 18 montre les bibliothèques requises.

```
# Importer les bibliographie
import tensorflow as tf
import tensorflow.keras
from tensorflow.keras.models import Model
from tensorflow.keras.layers import Input, Dense, Activation, Flatten, Conv2D, MaxPool2D, Dropout
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.losses import binary_crossentropy
from tensorflow.keras import metrics
import matplotlib.pyplot as plt
from sklearn.metrics import classification_report, confusion_matrix, plot_confusion_matrix, ConfusionMatrixDisplay
import numpy as np
```

Figure 18: Import des bibliographies

5.3.Diviser les données

Pour deviser les donnes en deux fichiers de traitement et de test, nous avons base sur la bibliothèque **splitfolders** avec un pourcentage de 80% pour le train et 20% pour le test. Étant donné que la dataset est de petite taille nous avons ignorer le pourcentage de validation.

```

▶ #Diviser les données
import splitfolders
Data= "/content/gdrive/MyDrive/covid/"
data_devided=("/content/gdrive/MyDrive/covid/devided")
splitfolders.ratio(Data , data_devided , seed=1337 , ratio = (0.8 , 0.0 , 0.2))

```

Figure 19: Division de données

5.4. Accès aux données

Après la division de données dans deux fichiers, il est nécessaire de l'accéder pour les utiliser. Alors nous avons créé des chemins d'accès comme montre la figure 20.

```

[30] # Create path
train_loc = '/content/drive/MyDrive/covid/devided/train/' #data_devided/train/'
test_loc = '/content/drive/MyDrive/covid/devided/test/'

```

Figure 20: Accès aux données

5.5. Data augmentation

La figure 21 montre la fonction écrite pour l'augmentation des données. La fonction prendra le chemin de toutes les images qui doivent être augmentées et écrira les images augmentées dans le dossier respectif. Les méthodes d'augmentation disponibles dans tensorflow sont appliquées.

Les images ont été accédées à l'aide d'ImageDataGenerator. Les images ont été chargées dans

le traitement et les tests avec ImageDataGenerator comme indiqué ci-dessous.

```
# resize images
trdata = ImageDataGenerator(rescale= 1./255,
                             shear_range = 0.3,
                             zoom_range = 0.3,
                             horizontal_flip = True,
                             )
traindata = trdata.flow_from_directory(directory = train_loc, target_size = (224,224))
tsdata = ImageDataGenerator(rescale= 1./255,
                              shear_range = 0.3,
                              zoom_range = 0.3,
                              horizontal_flip = True,
                              )
testdata = tsdata.flow_from_directory(directory = test_loc, target_size = (224,224))
```

Figure 21:Augmentation de données

5.6.Création de model

La figure 22 montre l'implémentation Python de l'architecture de CNN détaillée au début de ce chapitre.

```

▶ # definition de l'image d'entré
input_shape = (224,224,3)
# Crzation de model
# Input layer
img_input = Input(shape = input_shape, name = 'img_input')
# bloc_1
x = Conv2D(32, (3,3) , padding = 'same' , activation='relu', name = 'layer_1') (img_input)
x = Conv2D(64, (3,3) , padding = 'same' , activation='relu', name = 'layer_2') (x)
x = MaxPool2D((2,2), strides=(2,2), name = 'layer_3') (x)
x = Dropout(0.25)(x)
#bloc_2
x = Conv2D(64, (3,3) , padding = 'same' , activation='relu', name = 'layer_4') (x)
x = MaxPool2D((2,2), strides=(2,2), name = 'layer_5') (x)
x = Dropout(0.25)(x)
#bloc_3
x = Conv2D(128, (3,3) , padding = 'same' , activation='relu', name = 'layer_6') (x)
x = MaxPool2D((2,2), strides=(2,2), name = 'layer_7') (x)
x = Dropout(0.25)(x)

x = Flatten(name = 'fc_1')(x)
x= Dense(64, name = 'lyaer_8')(x)
x = Dropout(0.5) (x)
x = Dense(2, activation='sigmoid', name='predictions')(x)

```

Figure22: Architecture de model CNN

5.7.Génération de model

La figure 23 montre la génération de modèle on l'envoi les donnes des entre et les sortie.

```

▶ # Generer le model
model = Model(inputs = img_input, outputs =x , name='Binary_classification' )

```

Figure 23: Génération de model

5.8. Compiler le model

Pour compiler le model nous avons choisi l'optimisateur Adam, est un algorithme pour l'optimisation basée sur le gradient du premier ordre des fonctions objectifs stochastiques, basée sur des estimations adaptatives des moments d'ordre inférieur. La méthode est simple à mettre en œuvre, est efficace en termes de calcul, nécessite peu de mémoire, est invariante au redimensionnement diagonal des gradients et convient bien aux problèmes importants en termes de données et / ou de paramètres.

```
[35] # Compiler le model
      model.compile(optimizer='adam', loss=binary_crossentropy, metrics=['accuracy'])
```

Figure24: Compilation de model

5.9. Exécution le model

La commande décrite dans la figure 25 permet de lancer l'apprentissage en a comme paramètre notre images de train avec leur polarités, le nombre d'épochè ainsi les images pour la validation.

```
# start Train/Test
batch_size = 28
hist = model.fit(traindata, steps_per_epoch = traindata.samples//batch_size,
                 validation_data = testdata, validation_steps = testdata.samples//batch_size,
                 epochs = 50)
```

Figure 25: Exécution de model

6. Résultats et discussion

Afin de montrer les résultats obtenus pour notre modèle, on illustre dans ce qui suit les résultats en termes de précision et d'erreur ainsi que le rapport de classification et la matrices de confusion par rapport au nombre d'époques (Une époque décrit le nombre de fois que l'algorithme passe sur l'ensemble de données) pour chaque modèle.

6.1. Précision et erreur

Pour notre model en a utilisé le model CNN à l'aide de notre dataset et nombre d'époché de 50. Après l'analyse des résultats obtenus, On constate les remarques suivantes : D'après les Figures 26 et 27 La précision de l'apprentissage et validation augmente avec le nombre d'époque, ceci reflète qu'à chaque époque le modèle apprenne plus d'informations. De même, l'erreur d'apprentissage et de la validation diminue jusqu'à stable

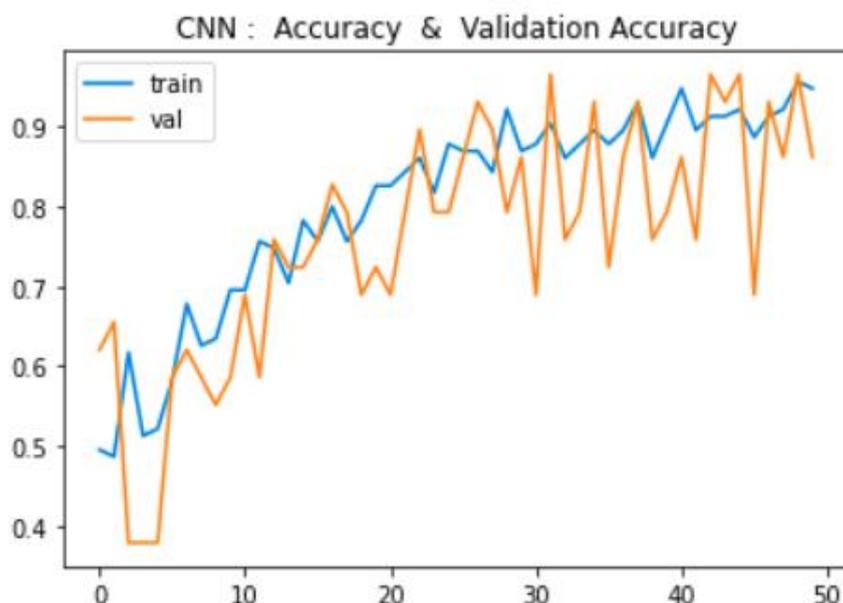


Figure 26: Précision de model avec 50 époques

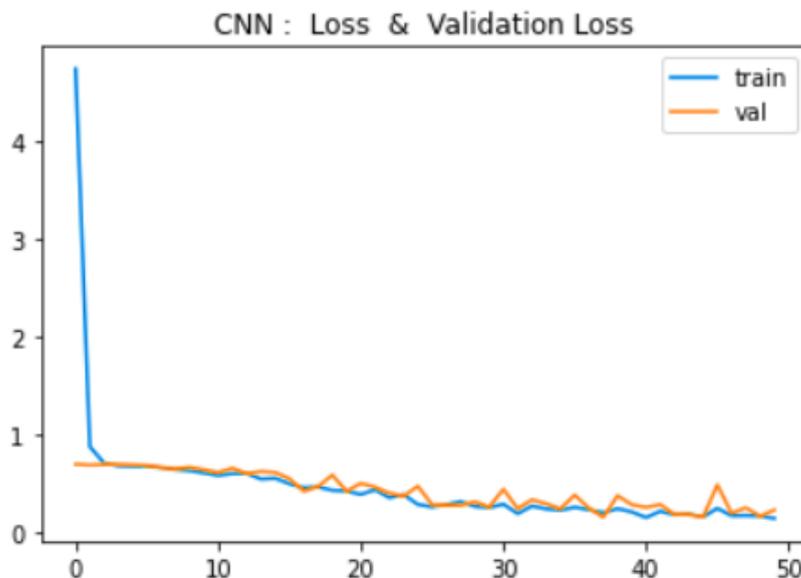


Figure 27: Erreur de model avec 50 époques

6.2. Le rapport de classification

Pour évaluer les performances d'un modèle de façon complète, vous devez analyser à la fois la précision et le rappel et le score.

✚ **La précision est le rapport** $\frac{TP}{TP+FP}$

Où tp est le nombre de vrais positifs et fp le nombre de faux positifs. La précision est intuitivement la capacité du classificateur à ne pas étiqueter comme positif un échantillon qui est négatif.

✚ **Le rappel est le rapport** $\frac{TP}{TP+FN}$

où tp est le nombre de vrais positifs et fn le nombre de faux négatifs. Le rappel est intuitivement la capacité du classificateur à trouver tous les échantillons positifs.

✚ **Le score F** peut être interprété comme une moyenne harmonique pondérée de la précision et du rappel, où un score F atteint sa meilleure valeur à 1 et le pire score à 0.

Les pondérations du score F rappellent la version bêta autant que la précision. $F = 1.0$ signifie que le rappel et la précision sont également importants.

✚ **Le support** est le nombre d'occurrences de chaque classe dans y_true .

✚ **Accuracy** $\frac{TP+TN}{TP+FP+FN+TN}$

La précision est la mesure de performance la plus intuitive et il s'agit simplement d'un rapport entre les observations correctement prédites et les observations totales. On peut penser que si nous avons une grande précision, notre modèle est le meilleur. Oui, la précision est une excellente mesure, mais uniquement lorsque vous avez des ensembles de données symétriques où les valeurs des faux positifs et des faux négatifs sont presque les mêmes. Par conséquent, vous devez examiner d'autres paramètres pour évaluer les performances de votre modèle.

$$\text{Accuracy} = \frac{13+12}{13+1+2+12} = 0.892857$$

La figure 28 montre le rapport de classification de notre modèle, nous remarquons que le recall pour la classe positive est égale à 0.93 et pour la classe négative égale à 0.86, ce qui signifie que le modèle fonctionne avec un pourcentage de 93 % avec la classe positive et 86 % avec la classe négative. Ce résultat est acceptable car le taux de biais est faible.

```

classification_Report
      precision  recall  f1-score  support
positive  0.86    0.93    0.90     14
negative  0.92    0.86    0.89     14

accuracy  0.90     28
macro avg 0.89    0.89    0.90     28
weighted avg 0.89    0.89    0.90     28

```

Figure28: Rapport de classification (précision ,Recall, F1-score)

6.3.Matrice de confusion

La matrice de confusion (matrice d'erreur), est un tableau qui présente différentes prévisions et résultats de tests, en les comparant avec des valeurs réelles.

Elles permettent d'effectuer plus rapidement une analyse approfondie des statistiques et de faciliter la lecture des résultats grâce à une visualisation claire des données.

Les résultats peuvent comprendre l'indication correcte d'une valeur positive comme étant un vrai positif ou d'une valeur négative comme étant un vrai négatif, ou encore d'une valeur positive incorrecte comme un faux positif ou d'une valeur négative incorrecte comme un faux négatif.

La matrice utilise des codes couleur donnent la possibilité de lire rapidement un certain nombre d'occurrences dans un test. Cette visualisation rapide permet d'analyser l'efficacité d'un modèle donné plus en détail qu'un résultat et de déterminer des tendances qui peuvent aider à régler les paramètres.

La matrice de confusion de notre modèle est présentée dans la figure 29.

Le modèle a bien classé 13 images positive par rapport 14 images , est 12 image négative par rapport 14 images.

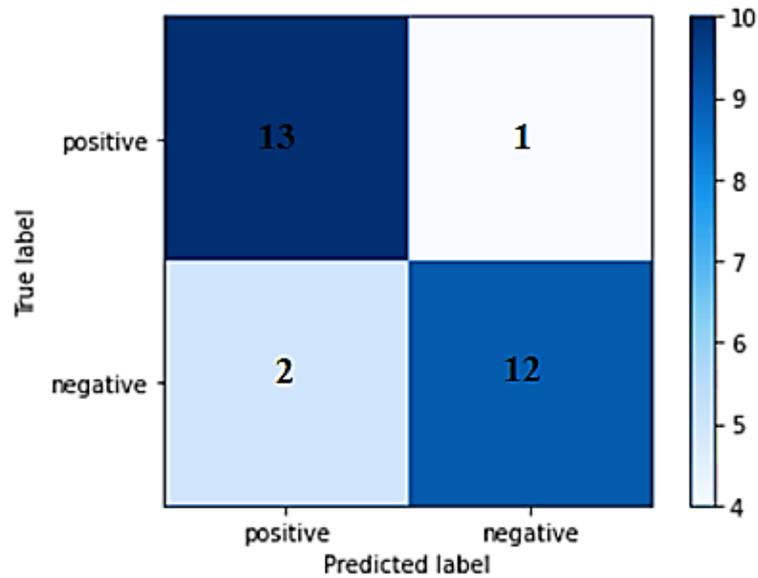


Figure 29 :Matrice de confison

6.4.Résultat de test

La figure 30 illustre le résultat de 28 enchantant de test.



Figure 30:Resultat de test

7. Conclusion

Nous avons présenté dans ce chapitre l'architecture globale de notre modèle de classification de la maladie de covid 19 , en première temps, nous avons détaillé notre contribution de classification des images commençant par la collection de données , le prétraitement de données , la création de modèle et finalement l'évaluation , ainsi nous avons détaillé les couches de modèle CNN créées , puis nous avons cité tous les outils hard et soft utilisés dans l'implémentation et nous finalisons le chapitre par les résultats obtenus et la discussion .

Conclusion

générale

1. Conclusion

Ces travaux de recherche s'inscrivent dans une contribution basée sur des algorithmes d'intelligence artificielle pour aider à diagnostiquer les maladies.

L'épidémie de COVID-19 a eu un impact profond sur le bien-être des personnes dans le monde et le nombre de décès liés à la maladie continue d'augmenter dans le monde. Bien que la technologie ait pénétré notre vie quotidienne avec un grand succès, en particulier en ML et DL, l'IA a également contribué à soutenir les gens dans la dure bataille contre le COVID-19.

DL n'est qu'un moyen potentiellement efficace de fournir des solutions prometteuses basées sur les données pour aider l'humanité à faire face au COVID-19.

Dans cette note, nous avons exploré les méthodes d'IA basées sur ML et DL pour diagnostiquer et traiter le COVID-19. De plus, dans la lutte contre le COVID-19, nous avons résumé les méthodes DL, les ensembles de données, les outils et les performances disponibles basés sur l'IA.

Nous avons commencé par fournir un aperçu complet de la maladie Corona, de ses types, de sa propagation et de la façon de la diagnostiquer. Ensuite, nous avons fourni une étude détaillée sur les dernières méthodologies et les applications actuelles des chercheurs de DL et de la communauté de la santé au sens large avec une description de la façon dont la DL et les données améliorent le cas de COVID-19, et d'autres études pour éviter l'épidémie de COVID-19. Les défis potentiels et les astuces lors de l'utilisation de DL sont également présentés. En guise de contribution à cette note, nous avons fourni un modèle détaillé basé sur l'algorithme de classification d'image le plus populaire "CNN" pour construire un système basé sur l'ensemble de données COVID-19 de Tebessa State.

Malgré la petite taille du jeu de données utilisé, les résultats obtenus sont très encourageants, car nous avons atteint un taux de 89% avec CNN construit à partir de zéro.

2. Perspectives

Nous n'avons travaillé que dans sa version initiale, et il est toujours ouvert pour des améliorations et des travaux de comparaison avec d'autres méthodes de classification.

Nos futures améliorations sont :

- 1. Développez notre ensemble de données en ajoutant plus d'images et en ajoutant plus de catégories et sous-catégories des types de maladie covid-19**
- 2. Appliquez des algorithmes CNN pré-entraînés comme Resnet et Alexnet pour augmenter et comparer les résultats et choisir le meilleur modèle.**
- 3. Appliquer l'approche de segmentation pour déterminer la zone du corps affectée par les covid-19**

Cela peut aider les médecins à accélérer le diagnostic de la maladie covid.

Références

bibliographies

REFERENCES

- [1] Samuel, A. L. (1959). Some studies in machine learning using the game of checkers. *IBM Journal of research and development*, 3(3), 210-229.
- [2] Simon, A., Deo, M. S., Venkatesan, S., & Babu, D. R. (2016). An overview of machine learning and its applications. *International Journal of Electrical Sciences & Engineering*, 1(1), 22-24.
- [3] Trask, A. W. (2019). Deep learning.
- [4] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, 521(7553), 436-444.
- [5] Deng, L., & Yu, D. (2014). Deep learning: methods and applications. *Foundations and trends in signal processing*, 7(3-4), 197-387.
- [6] : <https://fr.mathworks.com/discovery/deep-learning.html> . (Consulté le 20/03/2021)
- [7] : <https://www.saagie.com/fr/blog/qu-est-ce-que-le-deep-learning/> (le 20/03/2021)
- [8] Uijlings, J. R., Van De Sande, K. E., Gevers, T., & Smeulders, A. W. (2013). Selective search for object recognition. *International journal of computer vision*, 104(2), 154-171.
- [9] Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. *arXiv preprint arXiv:1506.01497*.
- [10] https://R%C3%A9seau_neuronal_convolutif (Consulté le 20/03/2021)
- [11] Kadous, D. Utilisation des réseaux de neurones comme outil du datamining: Génération de modèle comportemental d'un processus physique à partir de données.
- [12] Desir, C. (2013). Classification automatique d'images, application à l'imagerie du poumon profond (Doctoral dissertation, Université de Rouen).
- [13] : Covid-19 en Algérie (corona-dz.live)
- [14] : Review of analytical performance of COVID-19 detection methods | SpringerLink
- [15] : Hospitals Deploy AI Tools to Detect COVID-19 on Chest Scans - IEEE Spectrum

REFERENCES

- [16] :Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., Van Der Laak, J. A., Van Ginneken, B. and Sanchez, C. I. (2017). A survey on deep learning in medical image analysis, *Medical image analysis* 42: 60-88
- [17] : Imran, A.; Posokhova, I.; Qureshi, H.N.; Masood, U.; Riaz, S.; Ali, K.; John, C.N.; Nabeel, M. AI4COVID-19: AI enabled preliminary diagnosis for COVID-19 from cough samples via an app. arXiv 2020, arXiv:2004.01275
- [18] : Maghdid, H.S.; Ghafoor, K.Z.; Sadiq, A.S.; Curran, K.; Rabie, K. A novel ai-enabled framework to diagnose coronavirus covid 19 using smartphone embedded sensors: Design study. arXiv 2020, arXiv:2003.07434.
- [19]: Narin, A.; Kaya, C.; Pamuk, Z. Automatic detection of coronavirus disease (covid-19) using x-ray images and deep convolutional neural networks. arXiv 2020, arXiv:2003.10849
- [20] : Wang, L.; Wong, A. COVID-Net: A Tailored Deep Convolutional Neural Network Design for Detection of COVID-19 Cases from Chest X-ray Images. arXiv 2020, arXiv:2003.09871.
- [21]: Maghdid, H.S.; Asaad, A.T.; Ghafoor, K.Z.; Sadiq, A.S.; Khan, M.K. Diagnosing COVID-19 pneumonia from X-ray and CT images using deep learning and transfer learning algorithms. arXiv 2020, arXiv:2004.00038.
- [22] :Apostolopoulos, I.D.; Mpesiana, T.A. Covid-19: Automatic detection from X-ray images utilizing transfer learning with convolutional neural networks. *Phys. Eng. Sci. Med.* 2020, 43, 635–640.
- [23] : Sethy, P.K.; Behera, S.K. Detection of coronavirus disease (covid-19) based on deep features. Preprints 2020. 202003.0300.v1

REFERENCES

- [24] : Hemdan, E.E.D.; Shouman, M.A.; Karar, M.E. Covidx-net: A framework of deep learning classifiers to diagnose covid-19 in X-ray images. arXiv 2020, arXiv:2003.11055.
- [25] : Hall, L.O.; Paul, R.; Goldgof, D.B.; Goldgof, G.M. Finding covid-19 from chest X-rays using deep learning on a small dataset. arXiv 2020, arXiv:2004.02060.
- [26] : Apostolopoulos, I.D.; Aznaouridis, S.I.; Tzani, M.A. Extracting possibly representative COVID-19 Biomarkers from X-ray images with Deep Learning approach and image data related to Pulmonary Diseases. J. Med. Biol. Eng. 2020, 40, 462–469.
- [27] : Loey, M.; Smarandache, F.; Khalifa, N.E.M. Within the Lack of Chest COVID-19 X-ray Dataset: A Novel Detection Model Based on GAN and Deep Transfer Learning. Symmetry 2020, 12, 651.
- [28] :Maia, M.; Pimentel, J.S.; Pereira, I.S.; Gondim, J.; Barreto, M.E.; Ara, A. Convolutional Support Vector Models: Prediction of Coronavirus Disease Using Chest X-rays. Information 2020, 11, 548

