Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

UNIVERSITE ABDELHAMID IBN BADIS– MOSTAGANEM

Faculté des Sciences Exactes et d'Informatique

Département de Mathématiques et informatique

Filière : Informatique

MEMOIRE DE FIN D'ETUDES

Pour l'Obtention du Diplôme de Master en Informatique

Option : **Ingénierie des Systèmes d'Information**

Présenté par :

BENAHMED Sarah

ATTAR Latifa Meriem

THEME :

Intelligence artificielle au service des épidémies Application à la Covid19

Soutenu le : 04/07/2022.

Devant le jury composé de :

Mme. BENIDRIS F.Z	MCA	Université de Mostaganem	Président
Mme. BETOUATI F	MCB	Université de Mostaganem	Examinateur
Mme. HAMAMI Dalila	MCB	Université de Mostaganem	Encadreur

Année Universitaire 2021-2022

Résumé :

Dans la lutte contre les maladies infectieuses, il est important de comprendre comment les agents pathogènes infectieux se propagent parmi la population humaine, faisant de la modélisation épidémiologique un outil utile pour décrire le mécanisme de ces épidémies complexes afin d'interpréter les données épidémiologiques. Avec la nécessité d'estimer le nombre de cas de la COVID19 (infection, guérison ou décès), l'idée est de modéliser l'épidémie en utilisant l'une des techniques de l'intelligence artificielle, telles que les modèles à base d'agents, les modèles d'automates cellulaires, modélisation par algorithme génétique ou bien la modélisation par apprentissage en profondeur (Deep Learning). Dans notre travail nous allons adopter la méthode la plus utilisée dans les travaux existants qui est le Deep Learning appliqué à l'épidémie de la Covid19.

Mots-clés : épidémie, Covid19, technique de l'intelligence artificielle, modélisation, l'apprentissage en profondeur.

Abstract:

In the fight against infectious diseases, it is important to understand how infectious pathogens spread among the human population, making epidemiological modeling a useful tool to describe the mechanism of these complex epidemics to interpret epidemiological data. With the need to estimate the number of Covid19 cases (infection, recovery or death), the idea is to model the epidemic using one of the artificial intelligence techniques, such as agent based models, models of cellular automata, modeling by genetic algorithm or modeling by deep learning (Deep Learning). In our work, we will adopt the most used method in existing work, which is deep learning applied to the Covid19 epidemic.

Keywords : Epidemic, Covid19, artificial intelligence technique, modeling, Deep Learning.

Dédicace

Nous dédions Ce mémoire

A nos chers parents nos mères et nos pères

Pour leur patience, leur amour, leur soutien et leurs encouragements.

A nos frères et nos sœurs.

A nos amis et nos camarades.

Sans oublier tous les professeurs que ce soit du Primaires, du moyen, du secondaire ou de L'enseignement supérieur.

Sarah et Meriem

Remerciements

Avant tout nous remercions dieu le tout puissant qui nous a donné la force, la patience et le courage pour que nous puissions accomplir ce modeste travail.

Nous remercions profondément notre encadreur madame Hamami Dalila pour ses suivis, ses conseils, ses précieuses remarques et orientations dans notre travail.

Nous exprimons nos reconnaissances à toutes personnes qui ont contribué de près ou de loin à l'achèvement de ce travail, nos enseignants, nos amis, nos camarades.

Nous remercions également les membres de jury d'avoir accepté de juger ce modeste travail.

Liste des figures

Figure 1: L'évolution d'une épidémie avec ou sans mesures préventives. 4
Figure 2: Schéma de transmission du modèle SIR. 8
Figure 3: Schéma de transmission du modèle SIRS. 9
Figure 4: Schéma de transmission du modèle SEIR. 9
Figure 5: Schéma de transmission du modèle SEIRS. 10
Figure 6: Schéma de transmission du modèle SEIQR. 11
Figure 7: Coronavirus. 17
Figure 8: IA et Machine Learning et Deep Learning. 20
Figure 9: Exemple sur le fonctionnement du deep Learning. 21
Figure 10: Cellule LSTM 28
Figure 11: Opérations internes que l'on retrouve dans le LSTM 29
Figure 12: LSTM dispose de « 4 matrices de poids » (avec leurs biais) contrôlant ses portes 30
Figure 13: Fonctionnement de la porte d'oubli d'un LSTM 31
Figure 14: Fonctionnement de la porte d'entrée d'un LSTM. 32
Figure 15: Mise à jour de l'état de la cellule d'un LSTM. 33
Figure 16: Fonctionnement de la porte de sortie d'un LSTM. 34
Figure 17: Architecture de l'application. 38
Figure 18 : Architecture du LSTM 38
Figure 19: Page d'accueil. 40
Figure 20: Interface de chargement de données. 41
Figure 21 Message d'erreur 41
Figure 22: Interface de visualisation par tableau. 42
Figure 23: Interface de visualisation graphique des cas confirmés. 42
Figure 24: Interface de visualisation graphique des cas confirmés et guéris (tous). 43
Figure 25: Graphe de prédiction avec paramètres initiaux. 44
Figure 26: enregistrement avec paramètres initiaux. 45
Figure 27: Graphe de prédiction avec paramètres changés. 46

Figure 28: Enregistrement avec paramètres changés. 46
Figure 29: Consultation des paramètres. 47
Figure 30: Consultation des résultats en tableau. 47
Figure 31: Consultation des résultats en graphe. 48
Figure 32 : Graphe de prédiction avec paramètres initiaux 48
Figure 33 : Graphe de prédiction avec paramètres changés (taille de lot =2, époque=80, taux de validation =0.5)…. 49

Liste des tableaux

Tableau 1: valeurs des paramètres du SIR 40
Tableau 2: les valeurs des paramètres initiaux. 44

Liste des abréviations

Abréviation	Expression Complète
IA	Intelligence Artificielle
SIR	Susceptible, Infecté, Retiré/Récupéré
SIRS	Susceptible, Infecté, récupéré, Susceptible
SEIR	Susceptible, Exposé, Infecté, Retiré/Récupéré
SEIRS	Susceptible, Exposé, Infecté, Récupéré, Susceptible
SEIQR	Susceptible, Exposé, Infecté, Quarantaine, Récupéré
GA	Genetic Algorithm
ABM	Agent Based Modeling
CA	Cellular Automata
RNN	Recurrent neural network
CNN	Convolutional neural network
SIRVD-DL	Susceptible, Infected, Recovered, vaccinated, Dead, Deep Learning
LSTM	Long short-term memory
ANN	Artificial neural network
PC	Compartimentation de la population
LR	Low Riskcompartments
HR	High riskcompartments
ARIMA	Agressive Integrated Moving Average
OMS	Organisation mondiale de la santé
PLS	Partial least square

Table des matières

Introduction Général..... 1

Chapitre 1 Intelligence artificielle pour la modélisation épidémiologique 3

1.1. Introduction..... 3

1.2. Modélisation épidémiologique 3

1.2.1. Définition d'un modèle épidémiologique 4

1.2.2. Généralités sur les modèles utilisés en épidémiologie 5

1.2.3. Intérêt des modèles épidémiologiques..... 6

1.2.4. Paramètres de la modélisation 7

1.2.5. Mode de transmission 7

1.2.6. Modèles à compartiment..... 8

1.3. Techniques de l'intelligence artificielle pour la modélisation Épidémique 11

1.4. Conclusion 15

Chapitre 2 Modélisation avec Deep Learning pour la Covid19..... 16

2.1. Introduction..... 16

2.2. Epidémie de la Covid19..... 16

2.2.1. Description de l'épidémie de la Covid19 16

2.2.2. Son historique 17

2.2.3. Mode de transmission de la Covid19..... 18

2.2.4. Impact de la Covid19 sur la population..... 18

2.3. Deep Learning 19

2.3.1. Définition..... 19

2.3.2. Fonctionnement du Deep Learning 20

2.3.3. Algorithmes du Deep Learning 21

2.3.4. Utilités du Deep Learning..... 22

2.3.5. Champs d'application du Deep Learning..... 22

2.4. Modélisation de la Covid19 par le Deep Learning 23
2.5. Réseaux de neurones récurrents 26
2.6. Mémoire à long terme et à court terme 26
2.6.1. Mécanisme de fonctionnement de LSTM 27
2.7. Conclusion 34
Chapitre 3 Conception et développement 35
3.1. Introduction 35
3.2. Outils et techniques 35
3.2.1. Python 35
3.2.2. Anaconda 36
3.2.3. Jupyter Notebook 36
3.2.4. Keras 36
3.2.5. Tkinter 37
3.3. Architecture de l'application 37
3.4. Architecture du LSTM 38
3.5. Résultats et analyses 39
3.5.1. Jeu de données 39
3.5.2. Résultats 40
3.5.3. Analyse 49
3.6. Conclusion 50
Conclusion générale 51
Annexe 52
Bibliographies 53

Introduction Général

Contexte et objectifs

Aujourd'hui, les maladies infectieuses décrivent un problème primordial de santé publique. Devant l'allongement de la diffusion rapide de l'épidémie, la protection et le contrôle de la transmission de la maladie deviennent particulièrement importants. Vis-à-vis de cette menace, les chercheurs doivent se préparer à l'avance pour réagir rapidement et efficacement si une telle épidémie est déclarée. Cela nécessite une mise en œuvre des dispositifs de suivi et de prévention.

Afin de comprendre le mécanisme de propagation des maladies transmissibles, de nombreux scientifiques ont développé différents modèles. Le premier modèle en épidémiologie a été développé par Daniel Bernoulli en 1760 pour étudier le problème de l'inoculation de la variole.

L'épidémie de coronavirus a été découverte en Chine au mois de décembre 2019. Elle a bouleversé le monde entier par son évolution et sa transmission rapide. Selon un bilan établi à partir de sources officielles, ce phénomène a fait plus de 4 millions de morts dans le monde entier depuis fin 2019, et plus de 2 milliards de cas d'infection.

Grâce à l'intelligence artificielle plusieurs modèles sont apparus pour lutter contre ce phénomène qui s'amplifie. Les techniques de l'IA pour la modélisation épidémiologique ont permis de :

- Étudier la façon dont les maladies se transmettent.
- Aider à se planifier pour lutter contre les maladies infectieuses.
- Ralentir ou empêcher la propagation des épidémies.

L'urgence de développer des modèles efficaces pour estimer la propagation de la pandémie du covid19 est devenue une nécessité essentielle. Les techniques de l'intelligence artificielle sont exploitées pour estimer le nombre des cas infectés, guéris et décédés, bien plus, ces techniques peuvent nous aider à comprendre ce qui pourrait se passer dans certains scénarios. Elles aident les chercheurs à simuler, à planifier et à agir pour obtenir le meilleur résultat possible. Dans ce contexte nous utiliserons le Deep Learning pour atteindre les objectifs suivants :

- Prévoir efficacement l'évolution de la maladie en estimant correctement les nouveaux cas confirmés et guéris.
- Extraire les caractéristiques relatives à la Covid19 tel que les paramètres nécessaires à la modélisation.
- Valider le mode de propagation de cette épidémie.

Organisation du mémoire

Ce mémoire est structuré comme suit :

Chapitre 1 : aborde les notions de base de la modélisation et de la simulation, les paramètres les plus utilisés dans la modélisation et les systèmes à compartiments, et résume les techniques de l'intelligence artificielle utilisées dans la littérature pour modéliser les épidémies.

*Chapitre 2 :*se compose de deux parties :

Partie 1 : définit la modélisation avec Deep Learning et son application à l'épidémie de la covid19.

Partie 2 : définit le concept de la mémoire à long terme et à court terme (LSTM) et son fonctionnement.

*Chapitre 3 :*aborde les différents outils et techniques utilisés à la réalisation de l'application, l'architecture de l'application, l'architecture du modèle LSTM ainsi que les résultats et analyses.

Chapitre 1

Intelligence artificielle pour la modélisation épidémiologique

1.1. Introduction

L'épidémie du coronavirus 2019 (Covid19) a posé un défi sans précédent à l'humanité, avec plus de 3,2 millions de cas confirmés et un nombre de morts de plus de 225 000 jusqu'à fin avril 2020 dans le monde. La crise de la Covid19 illustre la nécessité d'une solide stratégie d'intelligence artificielle (IA). Les groupes de recherche en IA du monde entier travaillent de manière intensive pour s'attaquer à divers aspects de la crise de la Covid19, notamment épidémiologiques. Les techniques de l'IA pour l'épidémiologie telles que la modélisation à base d'agent ou les algorithmes génétiques et évolutionnaires sont fréquemment utilisées pour améliorer la précision des prédictions pour le dépistage des maladies infectieuses et non infectieuses.

1.2. Modélisation épidémiologique

L'utilisation de la modélisation est importante afin d'aider à évaluer les activités de gestion des maladies. La valeur des modèles épidémiologiques réside dans leur capacité à étudier des scénarios de simulation et à fournir aux décideurs des éléments pour prédire les conséquences d'une invasion de maladies et l'impact des stratégies d'intervention. Bien qu'il s'agisse d'une partie importante, la prévision de l'évolution de la propagation du virus reste difficile, accompagnée de nombreuses incertitudes [1]. Dans le cadre de la lutte contre ce virus, la modélisation épidémiologique nous aide à comprendre la propagation du virus et nous permet d'évaluer les

effets des interventions pour limiter sa propagation, avec pour objectif principal d'aplatir la courbe épidémique pour la maintenir dans les limites hygiéniques des capacités du système comme le montre la figure 1 [2].

Il y a principalement trois grandes raisons de modélisation. La première raison est la description, cette partie descriptive met en jeu principalement des outils de la modélisation, où l'épidémie est décrite au mieux au vu des observations en tenant compte que la transmission du virus n'est jamais observée : on ne voit que des manifestations de l'épidémie. La deuxième raison est la compréhension, cela permet de dialoguer, de confronter les idées et les interprétations. La troisième raison est la prédiction, celle-ci conduit à contrôler plus efficacement la propagation de l'épidémie [3].

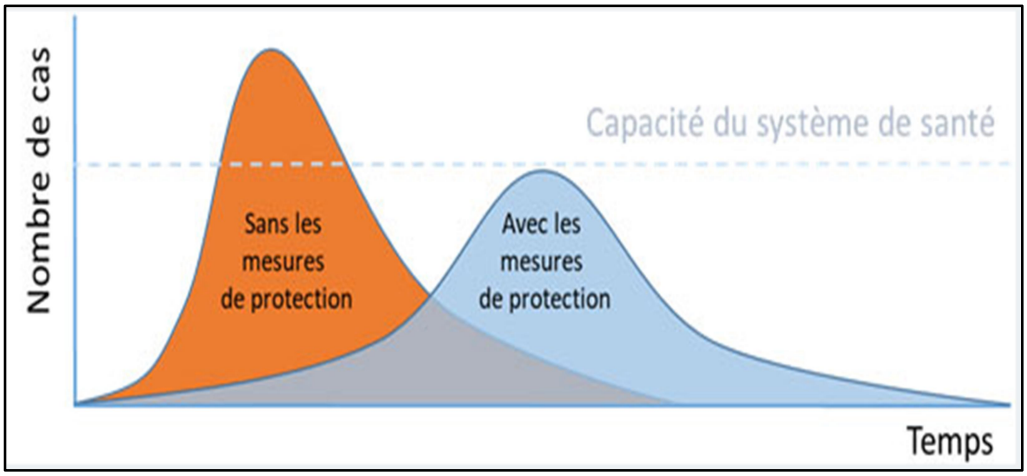


Figure 1: L'évolution d'une épidémie avec ou sans mesures préventives. [2]

1.2.1. Définition d'un modèle épidémiologique

Définir ce qu'est un modèle épidémiologique est très important. Les modèles épidémiologiques sont généralement définis comme des représentations épidémiologiques mathématiques et/ou logiques de la transmission des maladies et des processus associés. Ces modèles permettent de représenter la dynamique de propagation spatio-temporelle des maladies dans les populations concernées. Par conséquent, les modèles épidémiologiques peuvent aider à évaluer l'efficacité potentielle des interventions et fournir des estimations de leur ampleur et de leur durée. En ce qui concerne la gestion des maladies infectieuses, les modèles épidémiologiques

peuvent être définis plus largement, y compris une gamme de modèles statistiques/mathématiques, pas nécessairement limités à la mesure dans laquelle ils décrivent la maladie [1].

1.2.2. Généralités sur les modèles utilisés en épidémiologie

Essentiellement, tous les modèles sont des simplifications de systèmes plus complexes. Les modèles qui traitent de la maladie peuvent être divisés en différentes catégories en fonction de la façon dont ils traitent le caractère aléatoire (variabilité), le temps, l'espace et la structure de la population. Ces méthodes peuvent varier de simples modèles mathématiques déterministes à des simulations stochastiques spatialement explicites complexes. Le type de modèle le mieux adapté à une situation donnée dépend du problème étudié. Par conséquent, des modèles déterministes, de paramètres généralement basés sur des valeurs moyennes ou attendues, peuvent être utilisés pour comprendre la dynamique sous-jacente de l'infection. En revanche, ils ont une utilisation limitée en tant qu'outil prédictif car chaque pandémie est unique [4]. Mais lorsque nous avons des connaissances épidémiologiques et des données de haute qualité, nous pouvons développer des modèles plus raffinés pour décrire l'infection de différentes épidémies possibles.

En raison de la sophistication croissante des ordinateurs, associée à une plus grande prise de conscience de l'importance des facteurs spatiaux dans la transmission des maladies et à l'intérêt pour des stratégies de ciblage spatial spécifiques, les modèles qui intègrent des composantes spatiales prennent une importance croissante dans les études épidémiologiques [5]. De même, les modèles fondés sur les réseaux sont un domaine relativement récent mais en plein développement pour étudier, via des réseaux de contacts, la transmission des maladies [6].

Le processus de construction d'un modèle devrait commencer par une série de questions spécifiques conçues pour décrire la portée du modèle. Le choix du modèle repose sur sa compréhension, l'épidémiologie de la maladie, la quantité et la qualité des données disponibles et l'expérience du concepteur du modèle. Le niveau de sophistication qui doit être introduit dans un modèle est à la fois un art et une science. En effet, l'ajout d'éléments supplémentaires peut accroître la complexité sans nécessairement améliorer la qualité des résultats obtenus, tandis que l'ignorance de facteurs épidémiologiques apparemment importants peut conduire à des résultats trompeurs [4].

L’une des procédures critiques du développement d'un modèle est l’étape de vérification et de validation qui vise à garantir que ce modèle se comporte comme le système qu'il est censé représenter. La validation est le processus qui consiste à s'assurer que la logique, les formulations mathématiques et le code informatique d'un modèle reproduisent correctement le cadre logique développé par le concepteur du modèle [7].

La validation vise à s'assurer que le modèle est réaliste [7]. En d'autres termes, les hypothèses sur lesquelles il repose sont correctes et sa représentation du système étudié est raisonnable par rapport à l'objectif visé. Une approche plus holistique de la validité prend en compte : la validité des données, c'est-à-dire la précision des données utilisées pour construire et paramétrer le modèle ; la validité de construction, c'est-à-dire la précision de la logique mathématique et épidémiologique derrière le modèle ; et la validité opérationnelle, la capacité du modèle pour produire des résultats suffisamment précis lors de sa mise en œuvre.

1.2.3. Intérêt des modèles épidémiologiques

Le principal intérêt des modèles dans le domaine des épidémies est leur utilisation comme outils inter-épidémiques pour faciliter l’analyse rétrospective de virus et mieux comprendre leur comportement [8]. En combinant de grandes quantités d'informations de manière granulaire, on parvient à construire des scénarios hypothétiques pour comprendre les avantages de différentes stratégies dans différentes situations. Ainsi, les décideurs peuvent bénéficier de stratégies afin de diminuer la transmission contagieuse. La modélisation peut aider à améliorer le contrôle de la maladie [7] :

- En analysant les foyers antérieurs et évoluant des différentes stratégies de lutte.
- En recherchant différentes stratégies (élaboration de plans d'urgence) à employer en cas d'épidémie hypothétique.
- En étudiant les ressources nécessaires aux différentes stratégies de lutte contre une éventuelle épidémie (planification des ressources).
- En évaluant des risques, les domaines prioritaires, c'est-à-dire ceux associés aux risques potentiels peuvent être identifiés et les activités de préparation et de surveillance deviendront mieux ciblées.

- En évaluant l'efficacité des différentes stratégies de surveillance.
- En étayant des études d'impact économique.
- En fournissant des scénarios réalistes pour les exercices de formation par simulation et en communiquant l'épidémiologie et les mesures préventives de la maladie qui s'est déclarée.

1.2.4. Paramètres de la modélisation

Il existe plusieurs paramètres pour modéliser les épidémies. Parmi eux il y a ceux qui sont les plus fréquemment utilisés dans les travaux de la modélisation épidémiologique [12-15-18-46] tels que :

Le taux de reproduction : Le R0 est le chiffre qui représente le taux de reproduction moyen (ou la rapidité de contagion moyenne) d'une maladie, c'est-à-dire combien de personnes en moyenne un seul individu infecté risque de contaminer autour de lui [9].

Taux d'incubation : La période d'incubation couvre la période entre la contamination par le virus et l'apparition des premiers symptômes. Par exemple, la durée médiane d'incubation de la Covid-19 est de 5 jours. Cette durée est réduite avec la nouvelle variante Omicron [10].

Taux de transmission : Le taux de transmission est le rapport entre le nombre de personnes infectées lors d'un événement de contact étroit et le nombre de personnes lors de cet événement [10].

1.2.5. Mode de transmission

Nous avons distingué deux modes de transmission des maladies contagieuses : transmission par densité et transmission par fréquence. La transmission par densité est liée aux maladies infectieuses qui peuvent se propager par des gouttelettes respiratoires libérées dans l'air lorsqu'une personne infectée tousse ou éternue d'une proximité d'un mètre. La transmission par fréquence concerne les maladies sexuellement transmissibles et à transmission vectorielle ou la transmission à travers les liquides corporels (le sang, le lait maternel.) [11].

1.2.6. Modèles à compartiment

Les modèles compartimentaux séparent les populations en groupes et décrivent comment elles interagissent les unes avec les autres dans différents compartiments, le plus souvent, à l'aide d'équations différentielles. Il existe une diversité de modèles compartimentaux, nous présentons dans ce qui suit les modèles les plus connus :

Modèle SIR / SIRS

Kermack et McKendrick ont créé un modèle dans lequel chaque individu d'une population est affecté à un compartiment [12]. Au total, nous avons trois compartiments (comme le montre la figure 2) : S (Susceptible) pour les individus à risque d'infection, I (Infecté/Infectieux) pour les individus actuellement infectés et capables de transmettre la maladie à des sujets sensibles, R (Récupéré/Retiré) pour les personnes qui se sont rétablies de la maladie ou qui sont décédées. Ils ont supposé que tous les individus sont sensibles à la maladie et qu'une immunité complète est obtenue une fois l'infection guérie. Le modèle SIR est utilisé pour décrire une maladie, qui confère une immunité contre la réinfection.

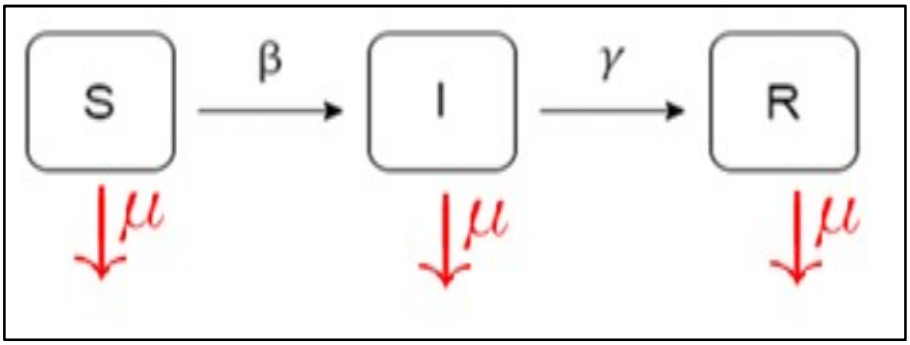


Figure 2: Schéma de transmission du modèle SIR. [12]

Il existe également le modèle SIRS. L'immunité d'une personne peut s'affaiblir avec le temps (immunité temporaire lors de la guérison d'une infection), de sorte que les personnes guéries retournent à un état sensible. La figure 3, illustre le schéma du modèle SIRS :

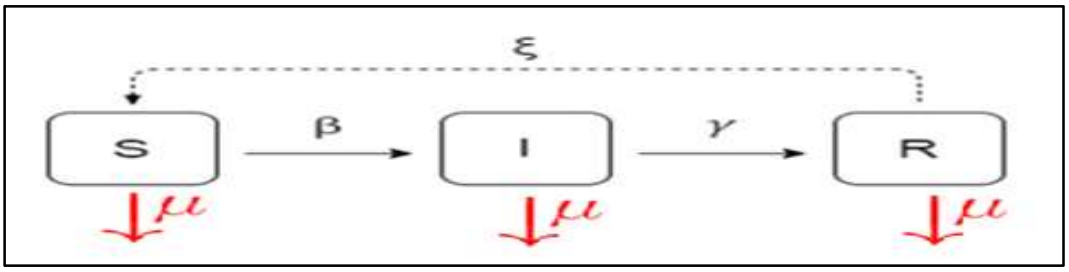


Figure 3: Schéma de transmission du modèle SIRS. [12]

Modèle SEIR / SEIRS

Le modèle mathématique SEIR [13] est largement utilisé dans le domaine de l'épidémiologie pour analyser les maladies infectieuses. Dans ce modèle, une nouvelle sous-population a été ajoutée : les personnes infectées non-infectieuses (exposed), et donc non contagieuse, représentée par la fonction E(t). Cela permet de considérer la période d'incubation de la maladie (via α le taux d'incubation). La figure 4 montre le schéma de transmission du modèle SEIR. Le modèle SEIRS a le même principe que le SEIR, mais les personnes guéries peuvent retourner à un état sensible (immunité temporaire) figure 5.

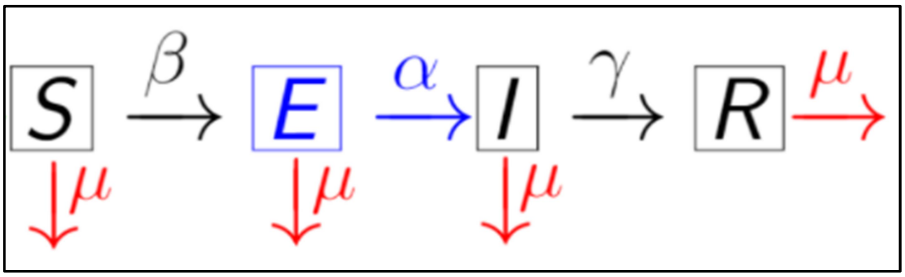


Figure 4: Schéma de transmission du modèle SEIR. [13]

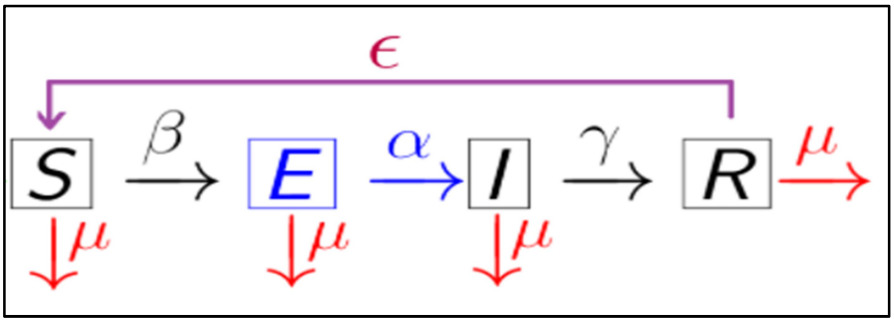


Figure 5: Schéma de transmission du modèle SEIRS. [13]

Modèle SEIQR

Certaines maladies sont traitées par la prévention (comme la vaccination) ou le traitement des patients qui présentent des symptômes. Pour les maladies qui n'ont pas de remède connu, la solution pour les contrôler et réduire leur transmission est d'isoler certains infectieux et de mettre en quarantaine les patients suspects. Au fil des ans, la quarantaine a été utilisée pour réduire la transmission de maladies telles que la peste, le SRAS, le choléra et la lèpre ou encore plus récemment la Covid19, dans des cas similaires, au lieu d'utiliser les modèles SIR ou SEIR, le plus efficace est SEIQR car il contient une classe pour les personnes en quarantaine (Q) [14].

Le SEIQR est un modèle amélioré de SEIR, SIR pour les maladies qui confèrent une immunité permanente. Le modèle SEIQR (comme le montre la figure 6) ne prend pas en considération la période de latence, après l'infection, l'individu sensible devient instantanément infectieux (classe I). Alors que dans le modèle SEIQR susceptible d'être exposé. Certains individus exposés restent dans la classe (E) pendant qu'ils sont en période de latence, puis passent à la classe supprimée (R) lors de la récupération. D'autres qui sont en fin de période d'incubation seront infectés (classe I). Les individus infectés peuvent récupérer directement ou après avoir été transférés dans la classe de quarantaine Q [14].

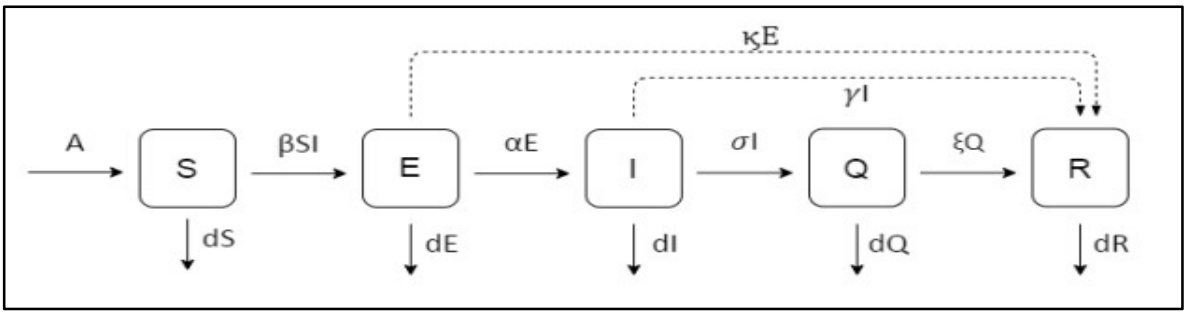


Figure 6: Schéma de transmission du modèle SEIQR. [14]

1.3. Techniques de l'intelligence artificielle pour la modélisation Épidémique

L'intelligence artificielle est importante pour la modélisation des épidémies. Elle joue un grand rôle pour détecter le groupe de cas et pour prédire où ce virus affecte à l'avenir. Elle aide les chercheurs à gérer les maladies et obtenir des suggestions appropriées en temps réel pour éviter leur propagation car l'IA fonctionne de manière efficace pour imiter l'intelligence humaine.

1.3.1. Modélisation avec les algorithmes génétiques

Les algorithmes génétiques (Genetics Algorithms : GA) sont des techniques de recherche et d'optimisation heuristique qui imitent le processus d'évolution naturelle. Les GA épidémiques simples opèrent sur une population de taille fixe d'individus de longueur fixe, et en général les individus sont représentés par une chaîne binaire qui encode les variables du problème que l'algorithme essaie de minimiser. Les GA simples utilisent essentiellement trois opérateurs : sélection, croisement et mutation [15].

L'algorithme génétique est une méthode d'optimisation puissante commune. M.Higazy et Maryam Ahmed Alyami [15], ont proposé une stratégie de contrôle optimale basée sur l'algorithme génétique, par laquelle les taux de contact et de mise en quarantaine optimaux peuvent être estimés. Fernando Concatto et al. [16], ont développé un nouvel algorithme génétique qui résout le problème nommé Min-SEIS-Cluster, un problème d'optimisation qui vise à minimiser la propagation de l'infection dans les réseaux. Ganesh Kumar M et al [17], ont appliqué le modèle de régression de croissance logistique et l'algorithme génétique pour prédire

le nombre de cas infectés par le coronavirus et ont également estimé la taille finale et son heure de pointe de l'épidémie de coronavirus en Inde. Mingxing Wu et al [18], ont proposé une nouvelle approche basée sur le modèle épidémique SIR et un algorithme génétique pour aider les concepteurs à trouver une combinaison de caractéristiques optimale qui maximise le capital client (customerequity). Zandkarimi et al [19], ont appliqué un GA pour sélectionner les caractéristiques les plus appropriées parmi plus de 1480 descripteurs de médicaments alcaloïdes. Ces ensembles de caractéristiques ont ensuite été extraits de médicaments connus pour former un réseau de neurones artificiels (ANN) afin de générer des modèles de prédiction relation quantitative structure-pharmacocinétique (QSPKR). Tan et al [20], ont mené une étude pour analyser la relation entre les éléments traces du sol et la mortalité par cancer du col de l'utérus en Chine. Une combinaison de GA et de méthode des moindres carrés partiels (Partial least square : PLS) a été utilisée pour choisir cinq des 25 oligo-éléments. Ensuite, un modèle de machine à vecteurs de support des moindres carrés (LSSVM) a été développé. LSSVM est une méthode utilisée dans l'apprentissage automatique pour déduire une fonction ou trouver un modèle dans les données d'apprentissage. Les résultats ont montré qu'une combinaison de GA-PLS et de LSSVM pouvait prédire la mortalité par cancer du col de l'utérus sur la base d'oligo-éléments.

Le concept des algorithmes génétiques est facile à comprendre, mais la mise en œuvre est encore un art. Les GA nécessitent moins d'informations sur le problème, mais concevoir une fonction objective et obtenir la représentation et les opérateurs corrects peut être difficile. Le GA est coûteux en temps de calcul.

1.3.2. Modélisation à base d'agent

Les modèles basés sur les agents (ABM) sont une classe de modèles informatiques qui fournissent une représentation à haute résolution, à la fois temporelle et spatiale, de l'épidémie au niveau individuel. Ces modèles permettent de prendre en compte plusieurs emplacements physiques, tels que les entreprises ou les écoles, ainsi que les caractéristiques uniques des communautés, telles que les tendances comportementales humaines ou les modèles de mobilité locale [21].

Une fois validés, les ABM peuvent être utilisés pour tester des scénarios de simulation concurrents qui nécessiteraient autrement des expériences peu pratiques et, potentiellement,

contraires à l'éthique. Par exemple, Ferguson et al [22], ont développé un ABM pour étudier l'impact des interventions non pharmaceutiques pour la Covid19, telles que le confinement à l'échelle nationale au Royaume-Uni et aux États-Unis. Aleta et al [21], ont proposé un ABM pour l'ensemble de la région métropolitaine de Boston afin d'élucider le rôle des différents types de mesures de recherche des contacts dans le maintien de faibles niveaux d'infection de coronavirus. Gressman et Peck [23] ont créé un ABM d'un campus universitaire pour examiner les stratégies de réouverture en toute sécurité des établissements d'enseignement supérieur. Hinch et al [24] ont formulé un cadre de modélisation open source basé sur des agents pour soutenir l'analyse de certaines interventions non pharmaceutiques et de programmes de recherche des contacts. Khaled M. Khalil et al [25], ont développé un modèle à base d'agents pour simuler la propagation de la grippe pandémique (nouveau H1N1) en Égypte. Liliana Perez et Suzana Dragicevic[26] ont développé une approche de modélisation à base d'agents qui intègre les systèmes d'information géographique (SIG) pour simuler la propagation d'une maladie transmissible en milieu urbain, à la suite des interactions des individus dans un contexte géospatial. Les mérites des ABM ont été reconnus par un grand nombre d'études, qui ont mis en lumière les aspects techniques de leur mise en œuvre ainsi que leur évolutivité à travers les scénarios.

Les modèles basés sur les agents sont une approche à faible coût et gain de temps, mais ils nécessitent des calculs sur supercalculateur, et nombreux paramètres, ce qui les rend difficile d'utilisation.

1.3.3. Modélisation avec Automate cellulaire

Les automates cellulaires (CA) ont été utilisés pour étudier la propagation des maladies contagieuses. Les CAs sont un système dynamique caractérisé par la discrétisation du temps et de l'espace. Habituellement, il consiste en une grille régulière de cellules connectées à leurs voisines et chaque cellule à un espace fini d'états [27].

Les CA ont été utilisés par plusieurs recherches comme une méthode alternative efficace pour simuler la propagation épidémique en dehors d'autres travaux parus dans la littérature des sciences de la vie et de l'informatique. Par exemple, Piotr Podolski et Hung Son Nguyen [27] ont utilisé les automates cellulaires dans la prédiction de Covid19. P.H.T.Schimit [28] a développé un modèle basé sur des automates cellulaires pour estimer l'impact de l'isolement social sur la

propagation de la Covid19 au Brésil. Jithesh PK [29] a utilisé des automates cellulaires pour étudier l'impact du confinement, de la migration et de la vaccination sur la dynamique de la Covid19. Ghosh S, Bhattacharya S. [30] se sont intéressés à la compréhension basée sur les données de la dynamique de la Covid19à l'aide d'automates cellulaires probabilistes basés sur un algorithme génétique séquentiel. Sayantari Ghosh et Saumik Bhattacharya [31] ont créé un modèle informatique sur la pandémie de la Covid19à l'aide d'automates cellulaires probabilistes. S. Hoya White et al [32],ont proposé un modèle théorique, basé sur des automates cellulaires, pour simuler la propagation épidémique. Plus précisément, il divise la population en trois classes : sensible, infectée et guérie, et l'état de chaque cellule représente la portion de ces classes d'individus dans la cellule à chaque pas de temps. M.A.Fuentes et M.N.Kuperman [33], ont présenté un modèle d'automates cellulaires développé pour étudier l'évolution d'un noyau d'infectiosité dans plusieurs conditions et pour deux types de maladies épidémiologiquement différentes.

Les automates sont capables de supporter de très grands espaces de paramètres pour la simulation, mais ils sont difficiles à interpréter analytiquement et nécessitent beaucoup de calculs et des grandes ressources pour le calcul.

1.3.4. Modélisation par apprentissage en profondeur (Deep Learning)

Deep Learning ou apprentissage en profondeur est utilisé dans différents domaines scientifiques. L'apprentissage en profondeur est une technique d'apprentissage automatique qui enseigne aux ordinateurs à faire ce qui vient naturellement aux humains. L'apprentissage en profondeur attire beaucoup l'attention ces derniers temps et pour une bonne raison. C'est d'obtenir des résultats qui n'étaient pas possibles auparavant, dépassant parfois les performances humaines.

Les modèles d'apprentissage en profondeur ont atteint des performances supérieures dans diverses analyses de classification, de régression et de séries chronologiques (Khan et al, 2020) [34]. Farooq et Bazaz [35], ont développé un modèle de réseau de neurones profonds (Deep neural network-modèle) de modélisation de la propagation de la Covid19 dans les cinq États les plus touchés de l'Inde. Jin et al [36], ont étudié les caractéristiques spatio-temporelles de la croissance de la Covid19 en Chine continentale et dans la province du Hubei. De plus, les

modèles d'apprentissage en profondeur ont montré des performances exceptionnelles dans la détection des cas de la Covid19 sur la base des données de radiologie thoracique. Par exemple, Aishwarya et Kumar (2021) [37] ont examiné et analysé l'application de diverses techniques d'apprentissage automatique et en profondeur pour tester la présence de la Covid19 à l'aide de radiographies pulmonaires et d'images de tomodensitométrie. Jayanthi Devaraj et al [38], ont étudié la fiabilité de la prédiction des cas de la Covid19 à l'aide de modèles Deep Learning. Yuexin Wu et al [39], ont développé un cadre d'apprentissage en profondeur pour prédire les profils épidémiologiques dans une perspective de séries chronologiques. Yuexin Wu et al ont adopté les réseaux de neurones récurrents (RNN) pour capturer la corrélation à long terme dans les données et les réseaux de neurones convolutifs (CNN) pour fusionner les informations provenant de données de différentes sources.

En plus de répondre aux mêmes critères offerts par les autres techniques, les modèles de Deep Learning offrent l'avantage d'utiliser de grandes quantités de données (BIG DATA) au contraire des autres techniques, ainsi que l'analyse et l'apprentissage des quantités massives de données sans surveillance, ce qui représente un outil précieux pour le Big Data Analytics où les données brutes sont en grande partie non marquées et non-classées.

1.4. Conclusion

Dans ce chapitre, nous avons présenté des notions de base sur la modélisation qui est un outil largement utilisé pour faciliter l'évaluation des activités de gestion des maladies, ensuite nous avons cité les paramètres les plus utilisés pour la modélisation épidémiologique, et puis nous avons également énuméré les systèmes à compartiment qui se basent sur le modèle SIR.

D'autre part, et suite à notre recherche sur les techniques de l'IA pour la modélisation épidémiologique telles que la modélisation à base d'agent, la modélisation avec les algorithmes génétiques...etc. Nous avons constaté que la modélisation par apprentissage en profondeur est la technique la plus performante pour modéliser le virus de la Covid19, car elle peut être adaptée à tout problème de modélisation, cette caractéristique lui offre un avantage majeur que d'autres techniques déjà vues. Nous allons voir plus en détails les performances du Deep Learning dans le chapitre qui suit.

Chapitre 2

Modélisation avec Deep Learning pour la Covid19

2.1. Introduction

La modélisation épidémiologique a été largement étudiée afin de contrôler la propagation des maladies. Elle se concentre sur la façon de quantifier avec précision la dynamique de transmission des maladies infectieuses. De nombreux chercheurs ont proposé différentes méthodes de prédiction pour établir un modèle de transmission du virus et prédire la tendance de la Covid19. Parmi elles, les méthodes basées sur l'intelligence artificielle qui sont actuellement les plus intéressantes et les plus utilisées telles que l'apprentissage en profondeur. La modélisation avec l'apprentissage en profondeur est largement utilisée dans l'analyse et la prévision de l'épidémie de la Covid19 et a de fortes capacités prédictives.

2.2. Epidémie de la Covid19

2.2.1. Description de l'épidémie de la Covid19

La covid19 du syndrome respiratoire aigu sévère est le septième coronavirus humain (figure 7), elle a été découverte à Wuhan, dans la province du Hubei, en Chine, lors de la récente épidémie de pneumonie en décembre 2019. La maladie est provoquée par un virus de la famille des Coronaviridae. Cette maladie infectieuse s'est rapidement propagée dans le monde entier [40].

Les patients infectés par ce phénomène peuvent présenter des symptômes allant de légers à graves surtout chez les âgés ou ceux qui ont des maladies chroniques, une grande partie de la population étant des porteurs asymptomatiques. Les symptômes les plus fréquemment rapportés sont la fièvre, la toux et l'essoufflement. Elle se propage par un simple contact ou rapprochement avec des personnes qui sont déjà infectées. Il existe des porteurs sains c'est-à-dire que les patients n'ont aucun symptôme apparent malgré la contamination [40].

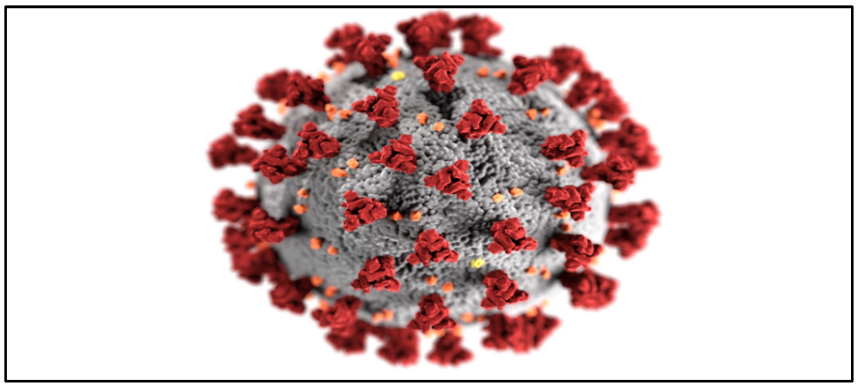


Figure 7: Coronavirus. [41]

2.2.2. Son historique

La pandémie de laCovid19 :

- Elle est causée par le coronavirus 2 du syndrome respiratoire aigu sévère (SARS-CoV-2).
- Elle est apparue fin 2019 (et donc son étiquette "2019" ou "19" dans son nom).
- A été identifié à Wuhan en décembre 2019 (bien que ce ne soit pas nécessairement son origine, qui reste à déterminer).
- A été déclarée comme émergence de santé publique de portée internationale par l'organisation mondiale de la santé en janvier 2020.
- A été déclarée comme pandémie (ou un statut élevé d'émergence de santé publique de préoccupation internationale) par l'Organisation mondiale de la santé en mars 2020.

- Elle est combattue à l'aide de vaccins pouvant être précommandés à partir de fin 2020 et distribués dans le monde entier à partir du début de 2021.

2.2.3. Mode de transmission de la Covid19

De nombreuses études empiriques ont tenté d'évaluer les effets des facteurs urbains sur la propagation des maladies infectieuses. Arbel et al. [42] ont évalué l'influence de la densité de population et des facteurs socio-économiques sur le taux d'infection à la Covid19 et ont constaté que la possibilité d'une infection à la Covid19 a augmenté avec les niveaux de densité de population. Ehlert [43] a tenté de trouver la relation entre densité et variables socio-économiques vis-à-vis du taux de mortalité. Il a signalé que la densité de la population était positivement corrélée avec les taux de mortalité dus à la Covid19. Wand et Li [44] ont découvert que la densité de la population était importante dans la propagation de la Covid19 au niveau des comtés américains, et qu'à cause de la densité de population ont cumulé jusqu'à 76% des cas d'infection aux États-Unis de début mars à fin Mai 2020.

2.2.4. Impact de la Covid19 sur la population

Contrairement à de nombreuses autres pandémies (par exemple, Ebola, H1N1, MERS, SRAS), la pandémie de la Covid19 a eu un impact sur le monde comme aucun autre - principalement en raison de sa présence mondiale. En ce sens, la pandémie de la Covid19 est similaire à la pandémie de grippe de 1918 (ou plus connue sous le nom de « grippe espagnole »), qui s'est produite entre 1918 à 1920, infectant environ un tiers de la population mondiale à cette époque (soit environ 500 millions de personnes), et tuant jusqu'à 100 millions de personnes. Cependant, la pandémie de la Covid19 est différente de la « grippe espagnole » car elle s'est produite à un moment où [45] :

• La quatrième révolution industrielle prolifère (par exemple, l'intelligence artificielle et l'internet des objets rendu possible grâce au big data et à la technologie blockchain), qui suggèrent que le monde est dans un état plus interconnecté que jamais

• La perturbation créée par la pandémie ne se limite pas à la santé publique (par exemple, des vies) mais également la santé économique (par exemple, les revenus, les emplois) du monde

• Les solutions peuvent être livrées avec une plus grande agilité à une plus grande population (par exemple, pandémie et vaccin mis au point la même année, soit 2020).

. La demande alimentaire et donc la sécurité alimentaire sont fortement affectées en raison des restrictions de mobilité, d'un pouvoir d'achat réduit et d'un impact plus important sur les groupes de population les plus vulnérables.

2.3. Deep Learning

2.3.1. Définition

L'apprentissage en profondeur ou "Deep Learning[1]" est un ensemble de méthodes d'apprentissage automatique (Machine Learning)[2] qui joue un rôle important en intelligence artificielle (figure 8). Cette approche a permis plusieurs avancées majeures où un programme peut tirer des conclusions sur des nouvelles données. En effet, le Deep Learning s'appuie sur un réseau de neurones artificiels qui est composé de milliers d'unités (les neurones) dont chacune recevant et interprétant les informations de la couche précédente par exemple : pour la reconnaissance visuelle, les couches supérieures identifient des formes, des combinaisons de formes, des objets, des contextes, etc. Les premières couches d'unités identifient des lignes, des courbes, des angles, etc. Ces techniques ont permis des progrès importants grâce au développement des ordinateurs et de grandes bases de données (big data). L'apprentissage en profondeur a été introduit pour rapprocher l'apprentissage machine de son objectif principal à savoir l'intelligence artificielle. Il concerne aussi des algorithmes d'apprentissage automatique qui essayent d'apprendre à plusieurs niveaux de représentation pour modéliser des relations complexes entre les données correspondant à différents niveaux d'abstraction. De plus, il a une bonne capacité d'extraire des caractéristiques à travers des données brutes grâce aux multiples couches de traitement qui sont composés de diverses transformations linéaire et non linéaire [46].

[1] Apprentissage en profondeur terme en anglais Deep Learning sera utilisé dans le reste du document.
[2] Apprentissage automatique terme en anglais Machine Learning sera utilisé dans le reste du document

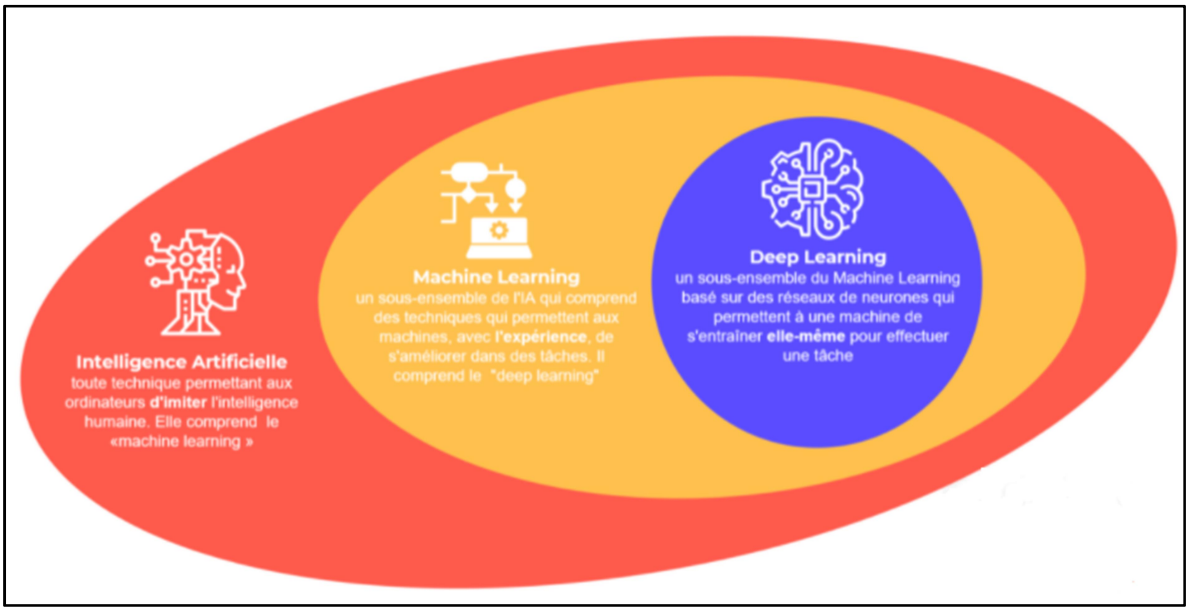


Figure 8: IA et Machine Learning et Deep Learning. [47]

2.3.2. Fonctionnement du Deep Learning

Le Deep Learning s'appuie sur un réseau de neurones artificiels qui s'inspire du cerveau humain. Le réseau de neurones est structuré comme le cerveau humain et se compose de neurones artificiels, également appelés nœuds. Ces nœuds sont empilés les uns à côté des autres en trois couches (La couche d'entrée, la ou les couches cachées, la couche de sortie). Les données fournissent à chaque nœud des informations sous la forme d'entrées. Le nœud multiplie les entrées avec des poids aléatoires, les calcule et ajoute un biais. Enfin, des fonctions non linéaires, également appelées fonctions d'activation, sont appliquées pour déterminer quel neurone déclencher. Par exemple, le système apprend à déterminer s'il y a un visage sur une photo avant de découvrir de quelle personne il s'agit, ou à reconnaître les lettres avant de s'attaquer aux mots dans un texte. À chaque étape, les « mauvaises » réponses sont éliminées et renvoyées vers les niveaux du début pour ajuster le modèle mathématique. Au fur et à mesure, les informations sont réorganisées en blocs plus complexes par le programme. Par la suite, ce modèle sera appliqué à d'autres cas, Les données de départ sont essentielles : plus le système accumule d'expériences différentes, plus il sera performant. À travers un processus d'auto-apprentissage, le Deep

Learning est capable d'identifier un chat sur une photo. Chaque couche du réseau neuronal est correspondante à un aspect particulier de l'image (figure 9) [48].

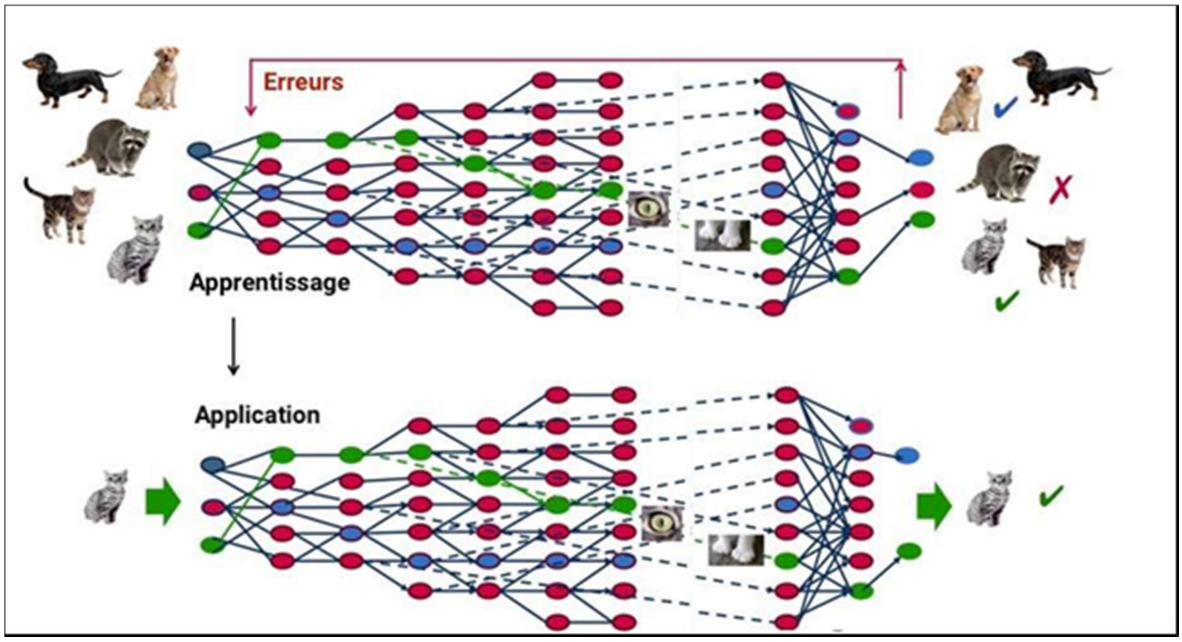


Figure 9: Exemple sur le fonctionnement du deep Learning. [48]

2.3.3. Algorithmes du Deep Learning

Un algorithme de Deep Learning est essentiel pour entraîner les machines dans le but d'exécuter des calculs complexes sur de grandes quantités de données. Il s'agit d'une catégorie de Machine Learning qui se base sur le fonctionnement d'un cerveau humain. Certains décrivent le Deep Learning comme un réseau de neurones multicouche. Mais pour être plus exact, il utilise différentes couches de réseaux pour traiter des données en fonction de leurs différents niveaux de caractéristiques. Autrement dit, les données sont transmises d'une couche à une autre pour affiner leur traitement et leur compréhension [49]. Parmi Les algorithmes d'apprentissage en profondeur y en a ceux qui sont les plus populaires tels que :

Réseaux de neurone récurrents (RNN)

Les réseaux de neurones récurrents sont conçus pour utiliser certains types de processus de mémoire artificielle qui peuvent aider ces programmes d'intelligence artificielle à imiter plus efficacement l'esprit humain. Les connexions entre les couches réseau dans les RNN forment des

boucles dirigées. En d'autres termes, cet algorithme d'apprentissage en profondeur utilise la sortie d'une couche comme une nouvelle entrée d'une autre couche. Les réseaux de neurones récurrents sont conçus pour sous-titrer des images, traiter le langage naturel ou être utilisés pour la traduction automatique et la modélisation épidémique. De plus, ils sont utilisés pour interpréter des informations temporelles ou séquentielles [49].

Réseaux de neurone convolutifs(CNN)

En apprentissage automatique, un réseau de neurones convolutifs (CNN ou ConvNet) est un réseau de neurones acycliques artificiels dans lequel les schémas de connexion entre les neurones sont inspirés du cortex visuel des animaux. Cette zone du cerveau est disposée de manière à correspondre à des zones qui se chevauchent lors de la mosaïque du champ de vision. Leurs fonctions s'inspirent des processus biologiques, et ils sont cohérents dans des empilements multicouches de perceptrons, mais où de petites quantités d'informations sont prétraitées. Les réseaux de neurones convolutifs ont un large éventail d'applications dans la reconnaissance d'images et de vidéos, les systèmes de recommandation, le traitement du langage naturel et la modélisation épidémique [46].

2.3.4. Utilités du Deep Learning

L'utilisation du Deep Learning offres les avantages suivants :

- L'apprentissage en profondeur fournit un cadre très souple, universel et pouvant être appris pour représenter des informations mondiales, visuelles et linguistiques.
- Peut apprendre à la fois sans surveillance et sous surveillance.
- Utiliser de grandes quantités de données d'entraînement.
- Extraire les caractéristiques de façon automatique.

2.3.5. Champs d'application du Deep Learning

L'apprentissage en profondeur s'applique à une variété d'industries, notamment :

- Reconnaissance visuelle, par exemple : reconnaissance des panneaux routiers par des robots ou des véhicules autonomes et parlants.
- La technologie robotique.

- La bio-informatique, par exemple pour l'étude de fragments non codants d'ADN et de génomes, ou le comptage cellulaire.
- Identifier ou comparer des formes.
- Sécurité.
- Santé.
- Méthode d'enseignement assistée par ordinateur.
- Art.
- Intelligence artificielle générale.
- Traduction.

2.4. Modélisation de la Covid19 par le Deep Learning

La Covid19 est devenue un grand défi auquel l'humain a été confronté ces derniers temps. De nombreux chercheurs ont proposé différentes méthodes pour constater le mode de transmission du virus et prédire l'évolution de la Covid19, parmi elles la technique d'apprentissage en profondeur qui est largement utilisée dans l'analyse et la prévision de l'épidémie de la Covid19 afin de réduire l'impact de ce virus sur l'humanité. Nous citons dans ce qui suit quelques travaux en relation :

Zhifang Lio et al, ont proposé un modèle de prédiction de la Covid19 basé sur le SIRVD en utilisant l'apprentissage en profondeur [50]. L'Objectif de ce modèle SIRVD-DL est d'évaluer l'évolution des paramètres de l'épidémie afin de construire le meilleur modèle pour prédire l'évolution tendancielle de l'épidémie. Ce modèle de prédiction SIRVD-DL est principalement divisé en trois parties : La première partie est le prétraitement des données : obtenir des ensembles de données sur la Covid19, y compris le nombre de cas confirmés, de cas récupérés, de cas décédés et de vaccination. Puis effectuer le prétraitement des données et transformer les données dans le format requis pour le modèle SIRVD-DL. La deuxième partie est la construction du modèle SIRVD : d'où un modèle SIRVD existant est modifié pour s'adapter aux changements dynamiques du temps. Ensuite, construire la courbe d'évolution des paramètres dans le temps en mesurant les paramètres du modèle. La dernière partie est le modèle d'apprentissage en profondeur : Sur la base de la mesure des paramètres du modèle obtenus à la dernière étape, la méthode de la moyenne mobile est utilisée pour lisser les données. Ensuite, un vecteur de série chronologique est construit comme entrée. La formation et l'évaluation des modèles sont réalisées grâce à des méthodes d'apprentissage en profondeur telles que mémoire à long et à court terme

(LSTM) et ses variantes pour sélectionner les meilleurs paramètres de modèle. Le modèle SIRVD-DL peut refléter les paramètres du modèle épidémique en temps réel dans la situation actuelle de la vaccination à grande échelle et prédire la tendance épidémique des maladies infectieuses. Ainsi, il peut évaluer les paramètres du modèle tels que le taux d'infection, le taux de récupération et le taux de mortalité. Le taux de précision des prédictions sur une journée s'améliore de 51 % par rapport aux meilleures méthodes de prédiction d'apprentissage en profondeur uniques existantes. Dans le même temps, le modèle fonctionne très bien dans les prévisions à court et à moyen terme. L'erreur de prédiction sur un jour est de 2,51 %, l'erreur de prédiction sur 7 jours est de 5,07 %, l'erreur de prédiction sur 14 jours est de 10,93 et l'erreur de prédiction sur 28 jours est inférieure à 30 %.

Junaid Farooq et Mohammad Abid Bazaz ont utilisé l'apprentissage en profondeur pour proposer un algorithme d'apprentissage incrémental en temps réel basé sur un réseau de neurones artificiels (ANN) et guidé par un flux de données pour l'estimation des paramètres d'un modèle analytique non intrusif, intelligent, adaptatif et en ligne de la maladie Covid19 [51]. Leur principale contribution est que dans un scénario de données d'entraînement en constante évolution, contrairement aux techniques typiques d'apprentissage en profondeur, cet algorithme non intrusif élimine le besoin de recycler ou de reconstruire le modèle à partir de zéro chaque fois qu'un nouvel ensemble de données d'entraînement est reçu. Après validation du modèle, ils l'utilisent pour étudier l'impact de différentes stratégies de contrôle des épidémies. Enfin, les auteurs ont proposé et simulé une stratégie d'immunisation naturelle contrôlée par le biais d'une compartimentation de la population (PC) basée sur le risque, dans laquelle la population est divisée en compartiments à faible risque (LR) et à haut risque (HR) en fonction de facteurs de risque (comme les comorbidités et l'âge) et soumise à différentes dynamiques de transmission de la maladie en isolant le compartiment HR tout en permettant au compartiment LR de développer une immunité naturelle. À la sortie de l'isolement préventif, le compartiment HR se retrouve entouré d'un nombre suffisant d'individus immunisés pour empêcher la propagation de l'infection et ainsi la plupart des décès survenus dans ce groupe sont évités. Ils utilisent l'Inde comme étude de cas, où le taux d'infection β = 0,0811, taux de guérison γ = 0,0429 et taux de mortalité δ=0,0024 pour la période de confinement en Inde (du 24 mars au 8 juin 2020). Ils ont simulé des mesures préventives telles que le confinement, la vaccination et l'immunité collective pour

étudier leur impact sur l'évolution de la maladie Covid-19. Et proposé une méthode efficace pour réduire considérablement le nombre de décès causés par la pandémie au cas où un vaccin ne serait pas disponible au niveau de masse. Ce modèle peut être appliqué à n'importe quelle population dans le monde et serait un outil utile pour les décideurs, les responsables de la santé et les chercheurs pour améliorer l'efficacité de la prise de décision, la formulation des politiques et les prévisions.

Hyeontae Jo et al, ont établi une étude qui analyse la propagation de la covid19 en Corée du Sud à l'aide du modèle SIR [52]. L'objectif de cette étude est de définir un modèle de prédiction du nombre de reproductions plus précis en adoptant une approche d'apprentissage en profondeur pour le modèle SIR avec des paramètres dépendant du temps pour évaluer l'efficacité de l'intervention afin de freiner la propagation de la Covid19 en Corée du Sud. Le nombre de reproduction dépendant du temps, RTD est une mesure de la transmissibilité d'un agent infectieux. La R T D est une mesure importante pour évaluer si les efforts actuels sont efficaces ou si une intervention supplémentaire est nécessaire. Ce modèle a effectué des simulations pour trois provinces (Séoul, Busan et Daegu) et la Corée du Sud, en utilisant le nombre total de population N. Les données reposent sur le nombre cumulé de personnes testées (T), les cas confirmés (I, ou I pos), les cas négatifs (I neg) et les cas guéris ou décédés (R) du 7 février 2020 au 30 mars 2020 pour la Corée du Sud et du 5 mars 2020 au 30 mars 2020 pour les provinces administratives de Séoul, Busan et Daegu. R T D (t), le nombre de reproduction dépendant du temps, est défini par R T D (t) = β (t) /y (t). Cette étude a fourni les résultats de simulation pour cinq modèles NN : S net, I net, R net, β net et γ net pour S, I, R. Pour une évaluation plus précise des paramètres du modèle, ces chercheurs ont fourni des solutions numériques en utilisant les paramètres estimés. En utilisant ces paramètres dépendant du temps *β net* et *γ net*, les chercheurs ont constaté qu'ils peuvent approximer avec succès le modèle SIR tout en ajustant les données de la Covid19. De plus, ils trouvent que les paramètres *β net* et *γ net* sont suffisamment bons pour ajuster le modèle et les données.

Jayanthi Devaraj et al, ont étudié si la prévision des cas de la Covid19 à l'aide de modèles d'apprentissage en profondeur est fiable et pratiquement significative [53]. L'objectif de cette étude est d'effectuer une analyse comparative sur les modèles de prédiction tels que les approches ARIMA, LSTM, Stacked LSTM et Prophet pour prédire la croissance de la Covid19 concernant

le nombre d'individus infectés, le nombre de décès et le nombre de cas récupérés. Les précisions de prédiction doivent être comparées et, en conséquence, le modèle le plus approprié est sélectionné sur la base des différentes mesures de performance et par une analyse d'hypothèses statistiques. L'approche consiste à évaluer ces modèles pour en vérifier la fiabilité en les appliquant à un cas de croissance globale. Ensuite, une analyse prédictive spécifique à un pays (Inde) et à une ville (Chennai) est présentée comme une étude de cas dans une perspective de prévision en temps réel. La méthodologie proposée est utilisée pour prédire les cas d'infection à court et à moyen terme. Les résultats de l'analyse montrent que les modèles Stacked LSTM et LSTM ont surpassé les autres modèles étudiés avec une plus grande précision et cela prouve la fiabilité de la prédiction des cas de la Covid19.

Les différents travaux que venons de recenser, nous ont montré que l'épidémie de la covid est souvent modélisée en utilisant les algorithmes de DL de type récurrent, la raison pour laquelle nous allons dans ce qui suit nous attarder sur ce type d'algorithme**.**

2.5. Réseaux de neurones récurrents

Les réseaux de neurones récurrents sontun type de réseau neuronal artificiel utilisant des données séquentielles ou des données de séries chronologiques. Leur concept consiste à suivre une séquence d'informations. Évidemment, il faut connaître le premier mot pour prédire le mot suivant, car les deux sont interdépendants. Bien que les RNN sont très pratiques pour le traitement des données séquentielles, il s'avère qu'ils sont extrêmement difficiles à entraîner pour gérer la dépendance à long terme en raison du problème de la disparition du gradient, à cause de ce problème les chercheurs ont élargi la version des RNNs en une nouvelle version appelée LSTM [54].

2.6. Mémoire à long terme et à court terme

Long short term memory (LSTM) est une architecture de réseau de neurones récurrents (RNN) utilisé dans le domaine de l'apprentissage en profondeur (Deep Learning). À la différence des réseaux neuronaux à propagation avant, le LSTM a des connexions de feedback.

Le processus de transfert de données est le même que pour les réseaux de neurones récurrents standards. Cependant, les opérations de diffusion de l'information sont différentes. Au fur et à mesure que les informations transitent, les opérations décident quelles informations nécessitent un traitement supplémentaire et quelles informations sont supprimées. Les principales opérations comprennent les cellules et les portes. L'état de la cellule est utilisé comme canal de transmission d'informations. Vous pouvez considérer les cellules comme des souvenirs.

Il existe plusieurs portes dans le processus LSTM qui facilitent le flux de nouvelles informations lorsque l'état de la cellule transporte des informations. Gates indique quelles données sont utiles et lesquelles ne le sont pas, ce qui permet de les rejeter. Par conséquent, seules les données pertinentes de la chaîne de séquence peuvent passer pour faciliter la prédiction [55].

2.6.1. Mécanisme de fonctionnement de LSTM

Il s'agit d'une architecture RNN, ils s'efforcent de résoudre le problème des dépendances à long terme. Autrement dit, si l'état précédent qui influence la prédiction actuelle n'est pas dans un passé récent, le modèle RNN peut ne pas être en mesure de prédire avec précision l'état actuel [56].

Les LSTM sont des cellules dans les couches cachées du réseau neuronal. Une cellule de LSTM est composée de trois portes (Figure 10). Ces portes sont des zones de calculs qui contrôlent le flux d'informations nécessaires (en réalisant des actions spécifiques) pour prédire la sortie dans le réseau (on a également deux types de sorties nommées états) [57].

- Porte d'oubli (Forget gate)
- Porte d'entrée (Input gate)
- Porte de sortie (Output gate)
- État caché (Hidden state)
- État de la cellule (Cell state)

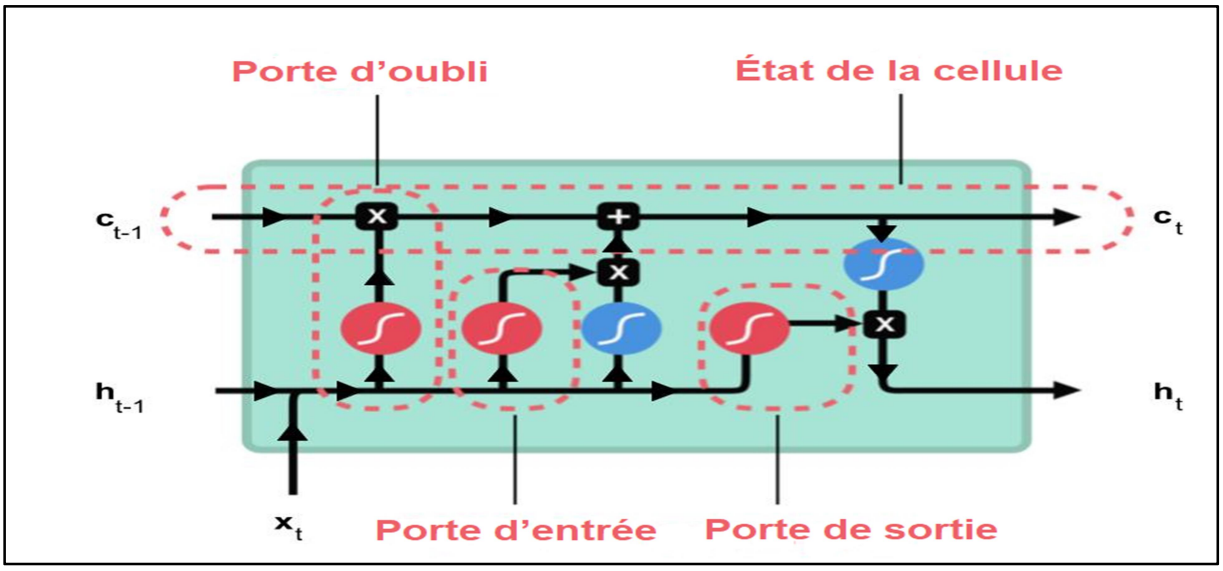


Figure 10: Cellule LSTM. [57]

Les opérations dans les portes permettent au LSTM de conserver ou supprimer des informations qu'il a en mémoire (Figure 11). Par exemple, dans la phrase « Hier soir j'ai mangé un hamburger et des », il est important de retenir les mots « hamburger » et « manger » tandis que les déterminants « un », « et » peuvent être oubliés par le réseau [57]. Les opérations possibles dans un LSTM sont :

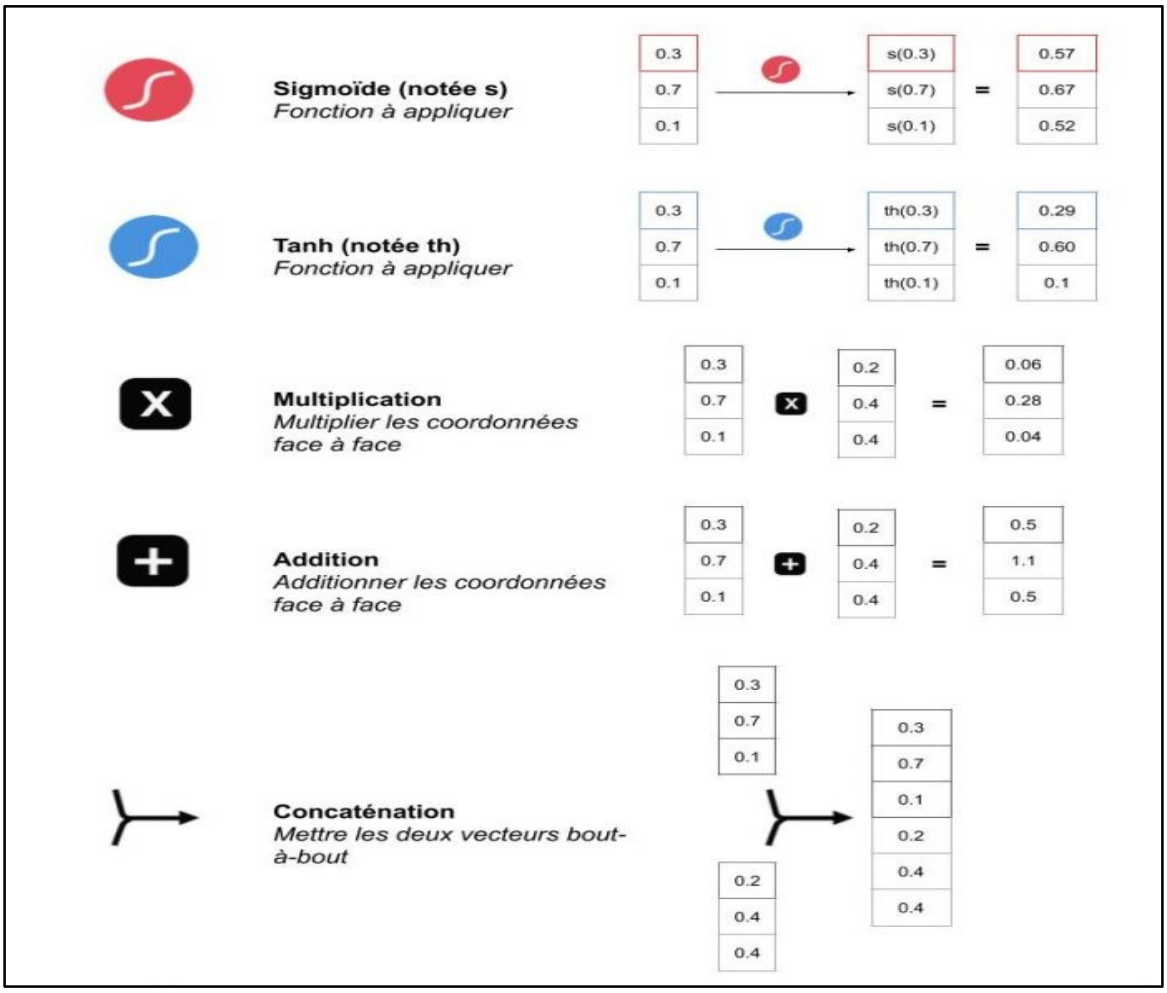


Figure 11: Opérations internes que l'on retrouve dans le LSTM [57]

Les données stockées dans la mémoire du réseau sont en fait un vecteur noté ct : l'état de la cellule. Comme cet état dépend de l'état précédent ct−1, qui lui-même dépend d'états encore précédents, le réseau peut conserver des informations qu'il a vues longtemps auparavant (contrairement au RNN classique) [57].

Les entrées de chaque porte sont pondérées par un biais et par des poids liés aux portes (Figure 12).On a 4 matrices de poids (leurs dimensions dépendent des dimensions de ht−1 et xt) :

- Wf: pondère l'entrée de la porte d'oubli (forgetgate)
- Wi : pondère l'entrée de la porte d'entrée (input gate)

- WC : pondère les données qui vont se combiner à la porte d'entrée pour mettre à jour l'état de la cellule (cell state)
- Wo : pondère l'entrée de la porte de sortie (output gate)

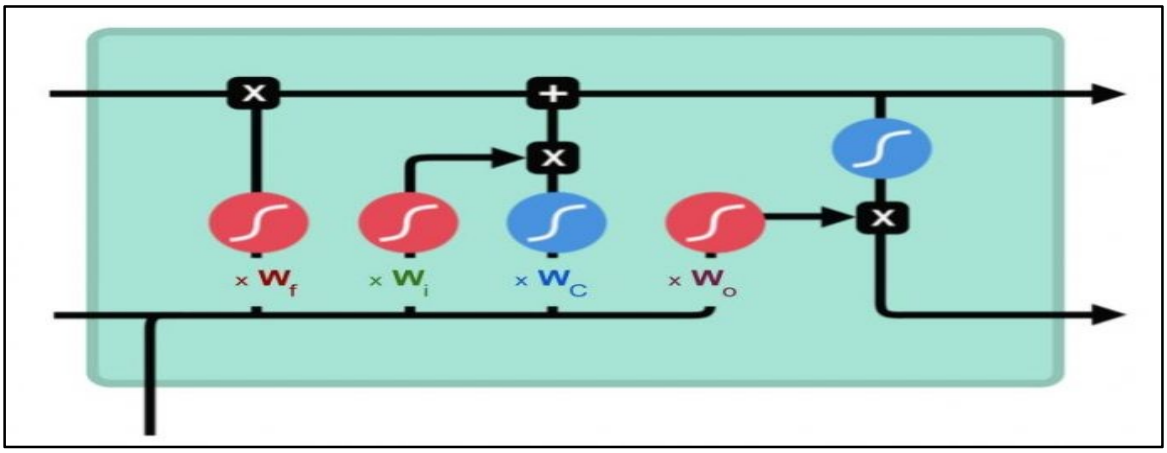


Figure 12: LSTM dispose de « 4 matrices de poids » (avec leurs biais) contrôlant ses portes [57]

Portes de la cellule LSTM

Détaillons à présent le fait de chaque porte, en gardant en mémoire que les données sont pondérées par les poids W (auxquels on ajoute un biais qui dépend de la porte et qui est aussi mis à jour dans la phase d'apprentissage) [57].

Porte d'oubli (forgetgate)

Cette porte décide de quelle information doit être conservée ou jetée : l'information de l'état caché précédent est concaténée à la donnée en entrée puis on y applique la fonction sigmoïde afin d'appliquer la normalisation en rendant les valeurs entre 0 et 1. Si la sortie de la sigmoïde s'approche de 0, cela signifie que l'on doit oublier l'information et si elle s'approche de 1 donc il faut la mémoriser pour la suite (Figure 13) [57].

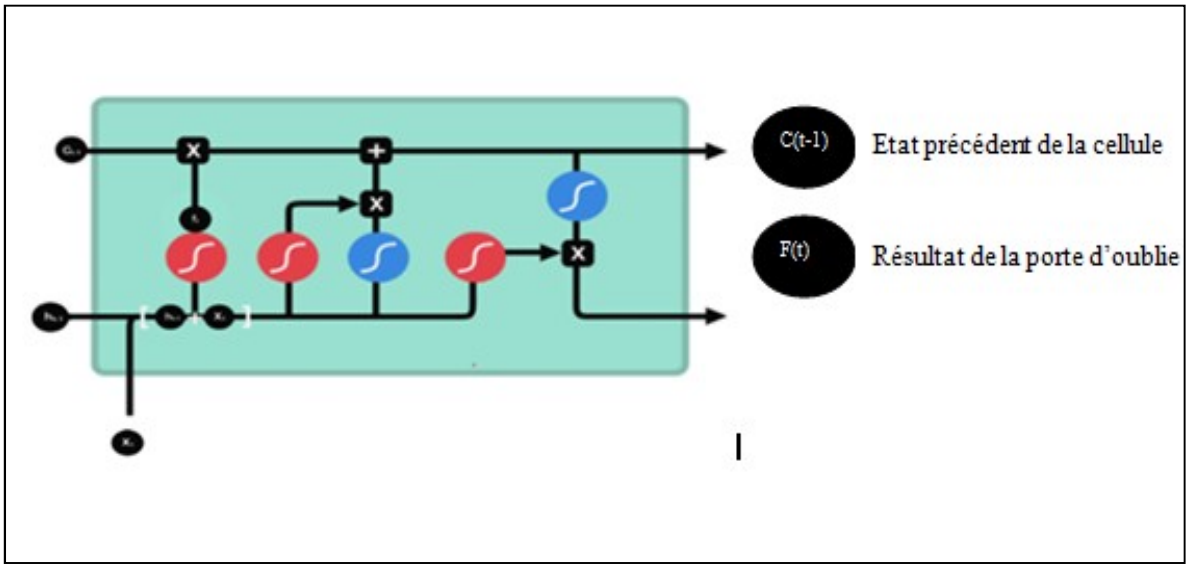


Figure 13: Fonctionnement de la porte d'oubli d'un LSTM [57]

Porte d'entrée (input gate)

La porte d'entrée (Figure 14) a pour rôle d'extraire l'information de la donnée courante : on va appliquer en parallèle une sigmoïde aux deux données concaténées (cf porte précédente) et une tanh [57].

- Sigmoïde va renvoyer un vecteur pour lequel une coordonnée proche de 0 signifie que la coordonnée en position équivalente dans le vecteur concaténé n'est pas importante. A l'inverse, une coordonnée proche de 1 sera jugée « importante ».
- Tanh va simplement normaliser les valeurs (les écraser) entre -1 et 1 pour éviter les problèmes de surcharge de l'ordinateur en calcul.
- Le produit des deux permettra donc de ne garder que les informations importantes, les autres étant quasiment remplacées par 0.

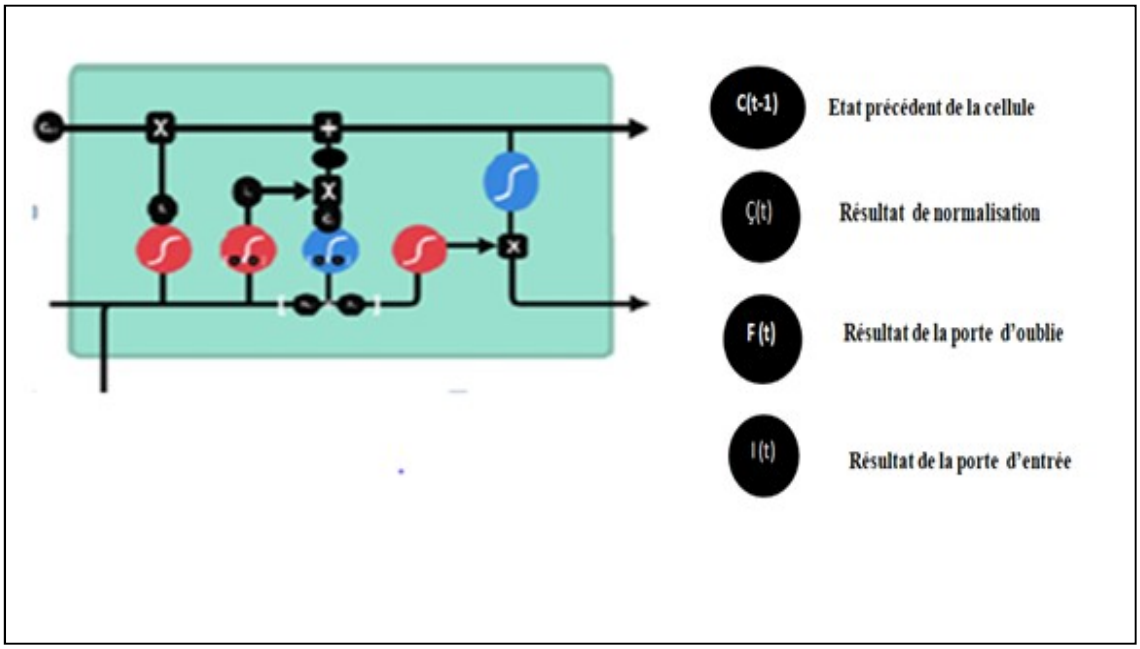


Figure 14: Fonctionnement de la porte d'entrée d'un LSTM [57]

État de la cellule (cell state)

L'état de la cellule se calcul assez simplement à partir de la porte d'oubli (Figure 15) et de la porte d'entrée : d'abord on multiplie coordonnée à coordonnée la sortie de l'oubli avec l'ancien état de la cellule. Cette opération permet d'oublier les informations de l'état précédent qui ne servent pas pour la nouvelle prédiction. Ensuite, on additionne le tout (coordonnée à coordonnée) avec la sortie de la porte d'entrée, en enregistrant dans l'état de la cellule ce que le LSTM (parmi les entrées et l'état caché précédent) a jugé pertinent [57].

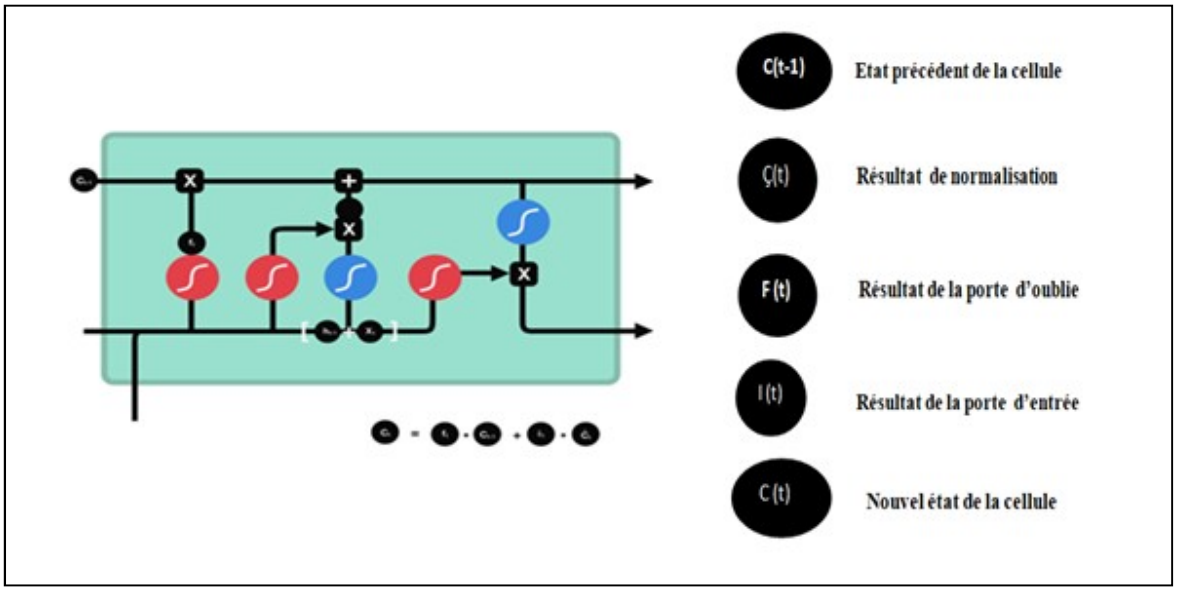


Figure 15: Mise à jour de l'état de la cellule d'un LSTM. [57]

Porte de sortie (output gate)

La porte de sortie doit décider quel sera le prochain état caché contenant des informations sur les entrées précédentes du réseau et qui sert aux prédictions (Figure 16). Pour cela, le nouvel état de la cellule calculé juste avant est normalisé entre -1 et 1 grâce à tanh. Le vecteur concaténé de l'entrée courante avec l'état caché précédent passé dans une fonction sigmoïde pour décider quelles informations à conserver (proche de 0 signifie que l'on oublie, et proche de 1 que l'on va conserver cette coordonnée de l'état de la cellule) [57].

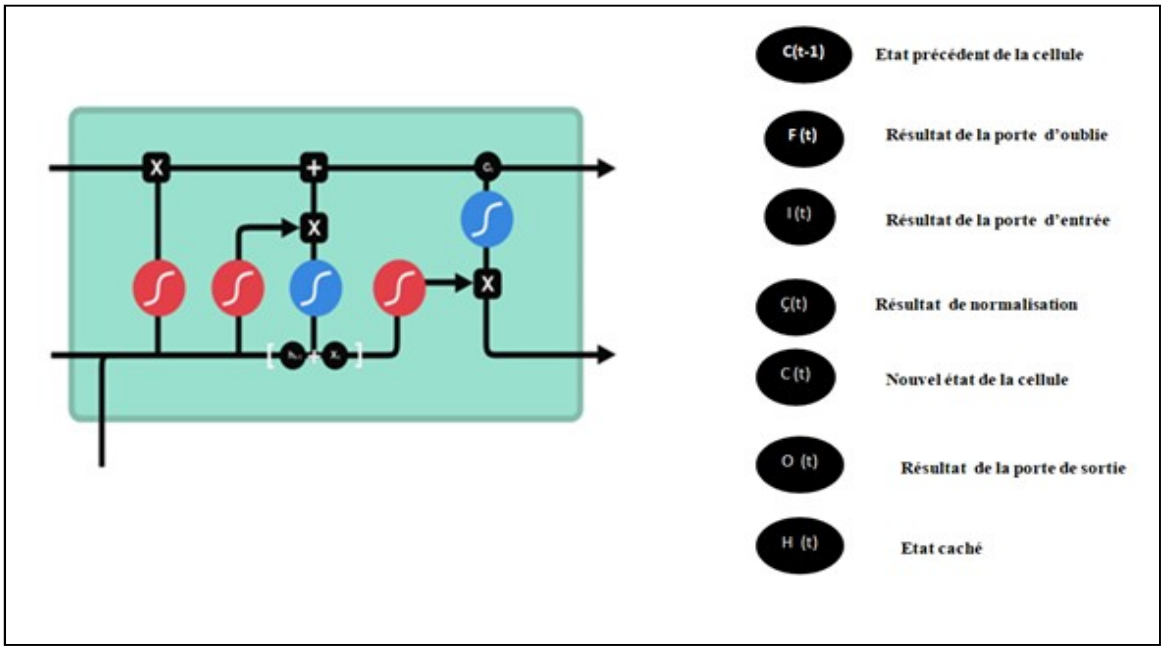


Figure 16: Fonctionnement de la porte de sortie d'un LSTM. [57]

2.7. Conclusion

Dans ce chapitre, nous avons décrit l'épidémie de la covid19 qui a bouleversé le monde ces derniers temps. Selon nos recherches nous avons distingué que la transmission de cette épidémie est par densité. D'une part, nous avons défini l'apprentissage en profondeur, son fonctionnement, son utilité et ses divers champs d'application. D'une autre part, nous avons mis en avant les travaux qui ont appliqué l'apprentissage en profondeur pour la modélisation de la covid19 afin de mieux comprendre l'intérêt de cette approche pour le contrôle et la compréhension de la Covid19. Cette technique permet d'étudier et de simuler l'évolution du processus de propagation de cette épidémie, et en dernier nous avons détaillé le fonctionnement de l'algorithme LSTM. Dans ce qui suit, nous allons utiliser le modèle LSTM que nous avons sélectionné à partir de la recherche bibliographique que nous avons effectuée, afin de prédire les cas futurs liés à cette épidémie. Nous avons choisi cet algorithme de Deep Learning (LSTM), car il a une plus grande précision que d'autres algorithmes considérés et peut fournir des résultats de prédiction fiables.

Chapitre 3

Conception et développement

3.1. Introduction

L'interaction entre l'humain et l'apprentissage en profondeur a été minutieusement enregistrée au cours des dernières décennies, et elle a encore été examinée à la lumière de la menace actuelle de pandémie mondiale de la COVID-19. Grâce à cette technique, nous pouvons mieux comprendre le mécanisme de la propagation de cette maladie.

Dans ce chapitre, nous allons voir quelques outils et techniques utilisées pour la réalisation de notre application, ainsi que l'algorithme de méthodologie de prévision que nous avons suivi et en dernier , nous allons aborder l'architecture de notre modèle LSTM avec les résultats et analyses que nous avons constaté durant cette expérience en expliquant l'intérêt de l'algorithme LSTM dans la prédictions des nouveaux cas confirmés et guéris.

3.2. Outils et techniques

Pour développer notre modèle, nous utiliserons plusieurs outils de test et d'analyse des données. Dans ce qui suit, nous présentons la définition des outils et techniques que nous avons utilisées pour réaliser ce projet :

3.2.1. Python

Python est un langage de programmation orienté objet, multiplateforme, open source et de haut niveau, c'est le plus utilisé par les informaticiens. Développer du code avec Python est simple et rapide, car il n'y a pas de contraintes de forme qui occupent du temps comme dans

d'autres langages, il prend en charge les modules et les packages, ce qui encourage la modularité du programme et la réutilisation du code.

Les principales utilisations de Python par les développeurs sont la programmation d'applications, l'apprentissage automatique, la science des données, la création de services Web, la génération de code, le Scripting et l'automatisation (interaction avec les navigateurs Web) [58].

3.2.2. Anaconda

Anaconda est un logiciel open source. Il facilite le développement de programmes python en science des données et en apprentissage automatique, c'est une boîte à outils qui vous permet de travailler avec des milliers de packages et de bibliothèques open source. A l'aide de plusieurs IDE, c'est la plate-forme de distribution Python la plus populaire car elle fournit des utilitaires pour créer, distribuer, installer, mettre à jour et gérer des logiciels de manière multiplateforme. Il facilite la gestion de plusieurs environnements de données qui peuvent être maintenus et exécutés séparément sans interférence les uns des autres [59].

3.2.3. Jupyter Notebook

Jupyter est une application web open source que vous pouvez utiliser pour créer et partager des documents contenant du code en direct, des équations, des visualisations et du texte. Il est livré avec le noyau IPython, qui vous permet d'écrire vos programmes en Python, mais il existe actuellement plus de 100 autres noyaux que vous pouvez également utiliser [60].

3.2.4. Keras

Keras est une bibliothèque open source de composants de réseaux neuronaux écrits en Python. Keras est capable de fonctionner sur TensorFlow, Theano, PlaidML et autres. La bibliothèque a été développée pour être modulaire et conviviale, mais elle a initialement commencé dans le cadre d'un projet de recherche pour le système d'exploitation intelligent neuro-électronique ouvert ou ONEIROS. Il est composé d'une bibliothèque de composants d'apprentissage automatique couramment utilisés, notamment des objectifs, des fonctions d'activation et des optimiseurs, la plate-forme open source de Keras offre également une prise en charge des réseaux de neurones récurrents et convolutifs. De plus, Keras propose le

développement de plates-formes mobiles pour les utilisateurs souhaitant mettre en œuvre des modèles d'apprentissage en profondeur sur les smartphones, à la fois iOS et Android [61].

3.2.5. Tkinter

Tkinter est une bibliothèque d'interface utilisateur graphique (GUI) open source et portable conçue pour être utilisée dans les scripts Python.Tkinter s'appuie sur la bibliothèque Tk, la bibliothèque graphique utilisée par Tcl/Tk et Perl, qui est à son tour implémentée en C. Par conséquent, on peut dire que Tkinter est implémenté en utilisant plusieurs couches [62].

3.3. Architecture de l'application

Dans cette section, nous allons voir l'architecture nécessaire pour développer l'application de prédiction pour prévoir les futurs cas de COVID-19 (Figure 17).

L'application nécessite des données COVID-19 comme données d'entrée pour prévoir les données futures correspondantes pour une période de temps définie à partir des données , un nettoyage des données et un processus de normalisation et partitionnement sont effectués,

Ensuite nous construisons le modèle LSTM et nous le validons. Après cette partie, une sortie se produit pour la prédiction et nous traçons les données (cas réels et cas prédit) dans un graphe pour voir le résultat.

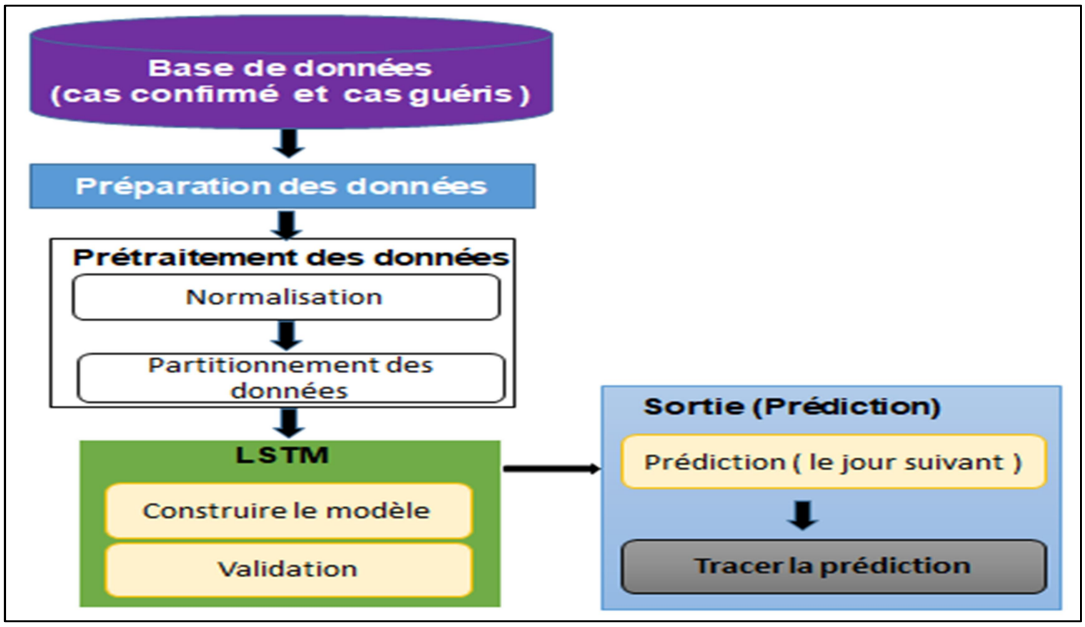


Figure 17: Architecture de l'application.

3.4. Architecture du LSTM

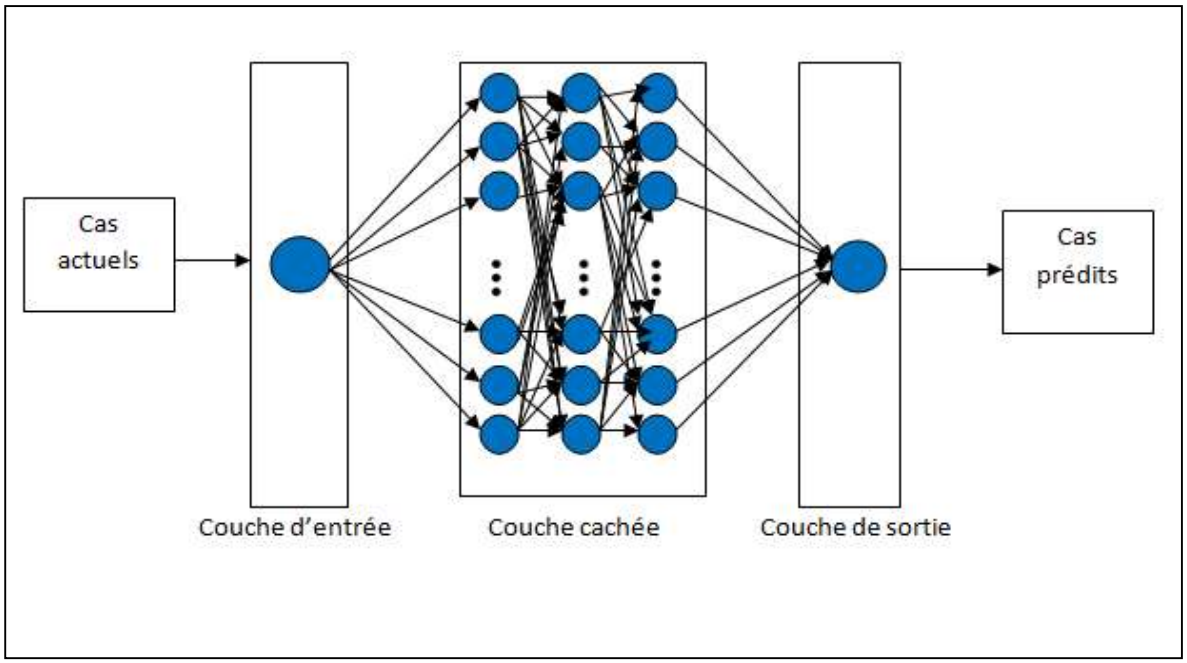


Figure 18 : Architecture du LSTM

Notre modèle LSTM (Figure 18) est structuré comme suite :

- Couche d'entrée : composée d'un seul neurone, car nous avons une seule entrée qui est les cas réels.
- Couche cachée : composée de trois couches LSTM. Nous avons augmenté le nombre de couche à 3 jusqu'à ce que nous ayons obtenu une bonne performance. Selon la bibliothèque keras, chaque couche se compose de 16 neurones ce qui équivaut à 48 neurones dans notre modèle.
- Les couches LSTM se basent sur une fonction« Dropout », cette dernière fait référence à la désactivation de neurones dans les couches pour éviter le sur-ajustement dans le modèle lors de l'entrainement.
- Couche de sortie : composée d'un seul neurone, car nous avons une seule sortie qui est les cas prédits.

3.5. Résultats et analyses

3.5.1. Jeu de données

Dans notre étude, nous avons utilisé l'ensemble des données extrait du travail de Dilip Kumar Bagal et al. [63] Où le modèle à compartiment de type SIR est appliqué sur les données avec les paramètres suivants:

β : représente le taux de transmission, γ : représente le taux de récupération et R0 : représente le taux de reproduction, leurs valeurs sont illustrées dans le tableau1. Pour reproduire les données réelles de la base de données en question, nous nous intéressons dans notre modèle aux cas infectés (confirmés) et récupérés (guéris) du pays inde (de 22 janvier 2020 au 30 avril 2020).

β	γ	R0
0.0622677	0.0850051	1.36515

Tableau 1: valeurs des paramètres du SIR.

3.5.2. Résultats

Au démarrage de l'application, le système affiche une interface représente la page d'accueil de notre application (Figure 19). A travers cette interface, nous donnons une description pour notre projet nommé « **Prévision COVID19** »avec une barre de menu composée de : données, visualisation et exécution**.**

Figure 19: Page d'accueil.

Charger

La figure ci-dessous représente la page de chargement des données, lorsque l'utilisateur appuie sur **données →charger**. Un menu contient différents fichiers de base de donnés ayant comme extension .csv s'affiche afin de charger les données sous sélection (Figure 20).

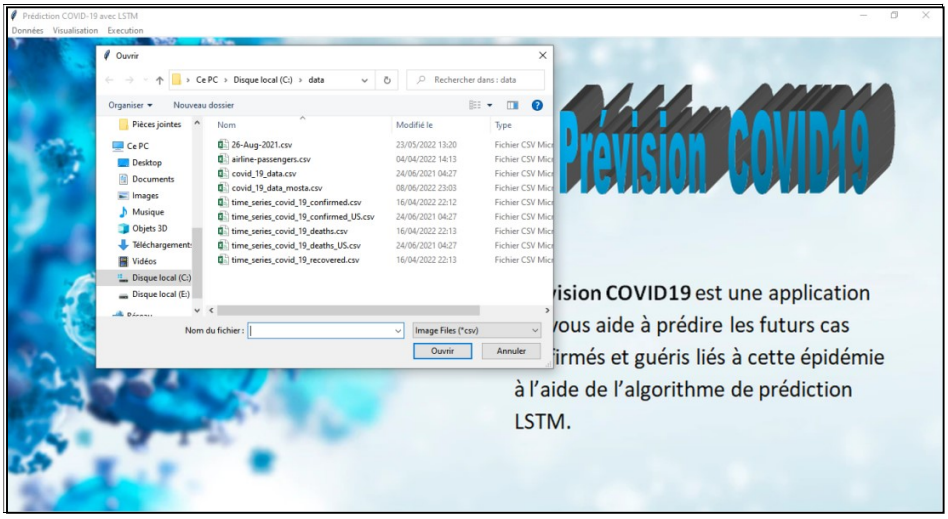


Figure 20: Interface de chargement de données.

Nous rappelons que les données doivent être structurées selon le schéma suivant : ObservationDate (date d'observation), Confirmed (confirmé), Recovered (guéris) etc. Si non, un message d'erreur (Figure 21) s'affiche indiquant que la base de donné introduite n'est pas relative au schéma déjà cité.

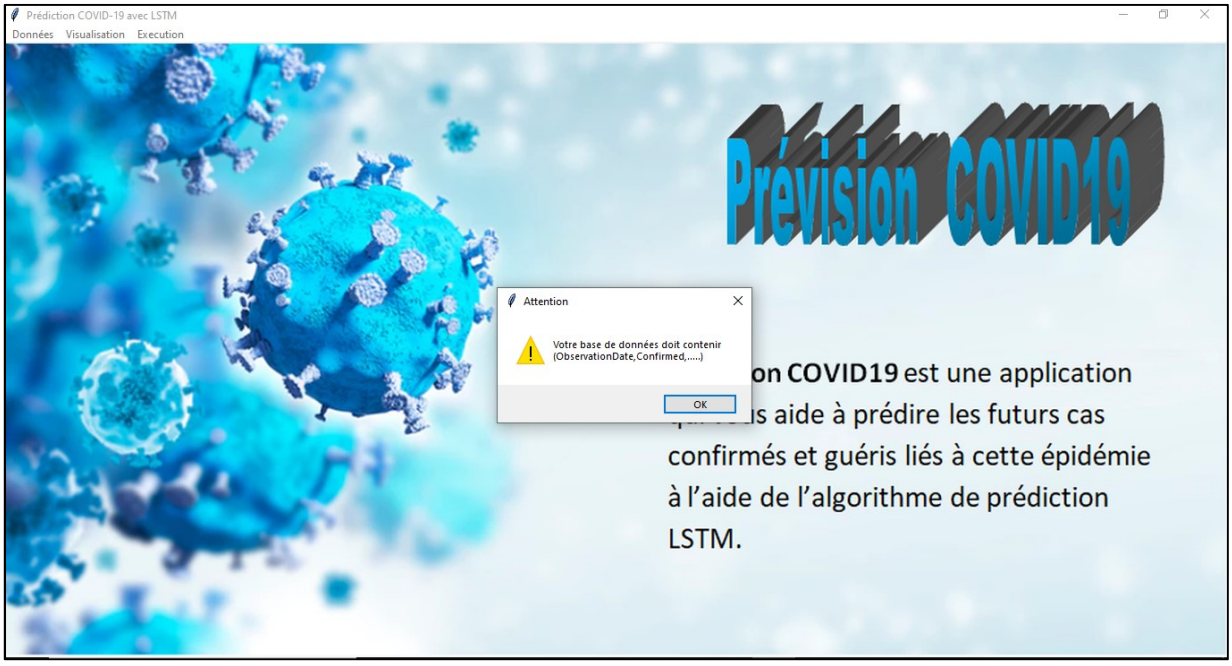


Figure 21: Message d'erreur.

Visualisation

Lorsque l'utilisateur charge une base de données, il peut visualiser le contenu de la base sélectionnée qui est structurée dans un tableau en appuyant sur **visualisation →par tableau** comme le montre la figure 22.

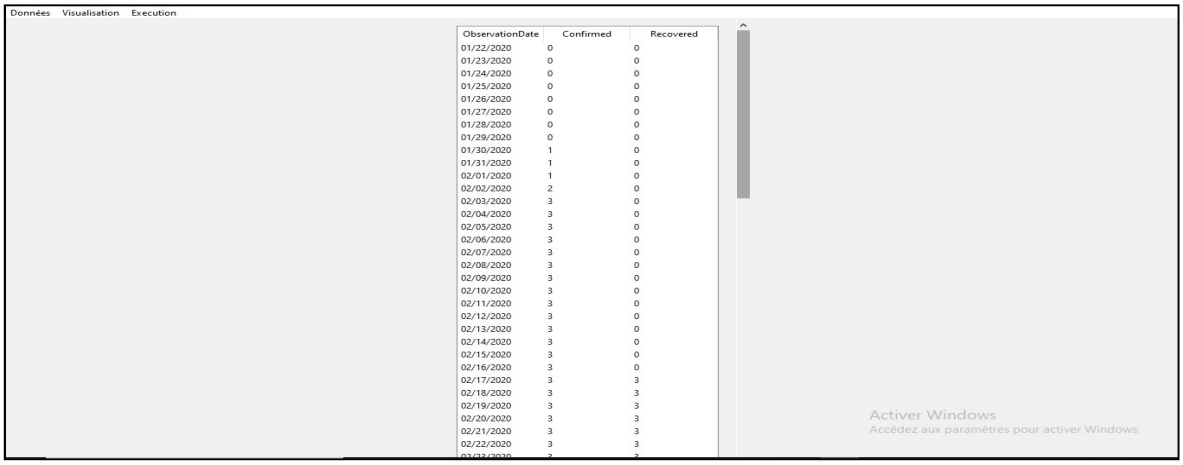


Figure 22: Interface de visualisation par tableau.

Comme il peut visualiser les données en graphe en appuyant sur **visualisation →graphique**, dans ce cas-là l'utilisateur peut choisir le type des cas à visualiser **→confirmé ou guéris ou bien tous** (confirmé et guéris et décédés) comme le montre les figures 23 et 24.

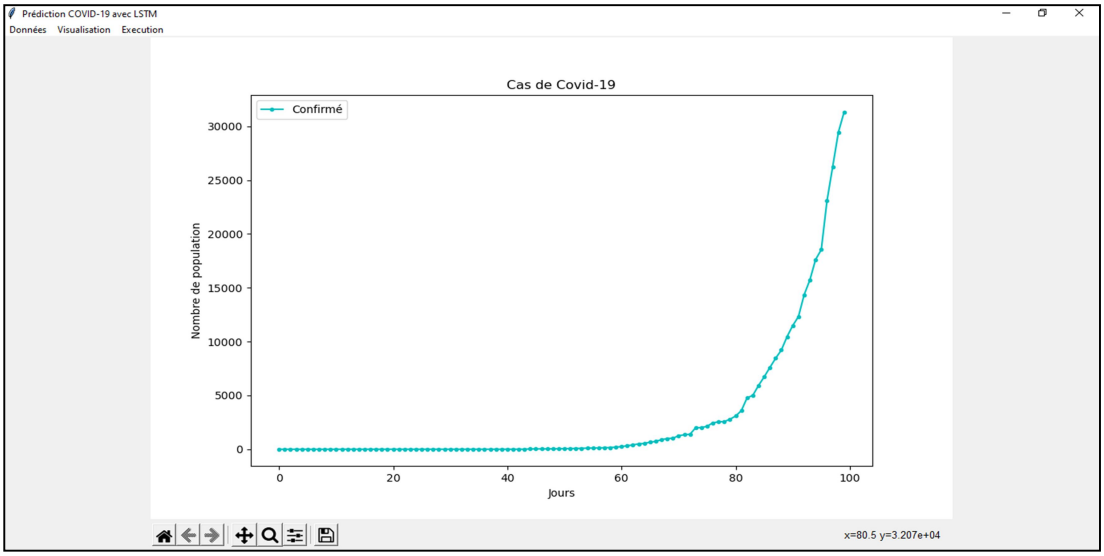


Figure 23: Interface de visualisation graphique des cas confirmés.

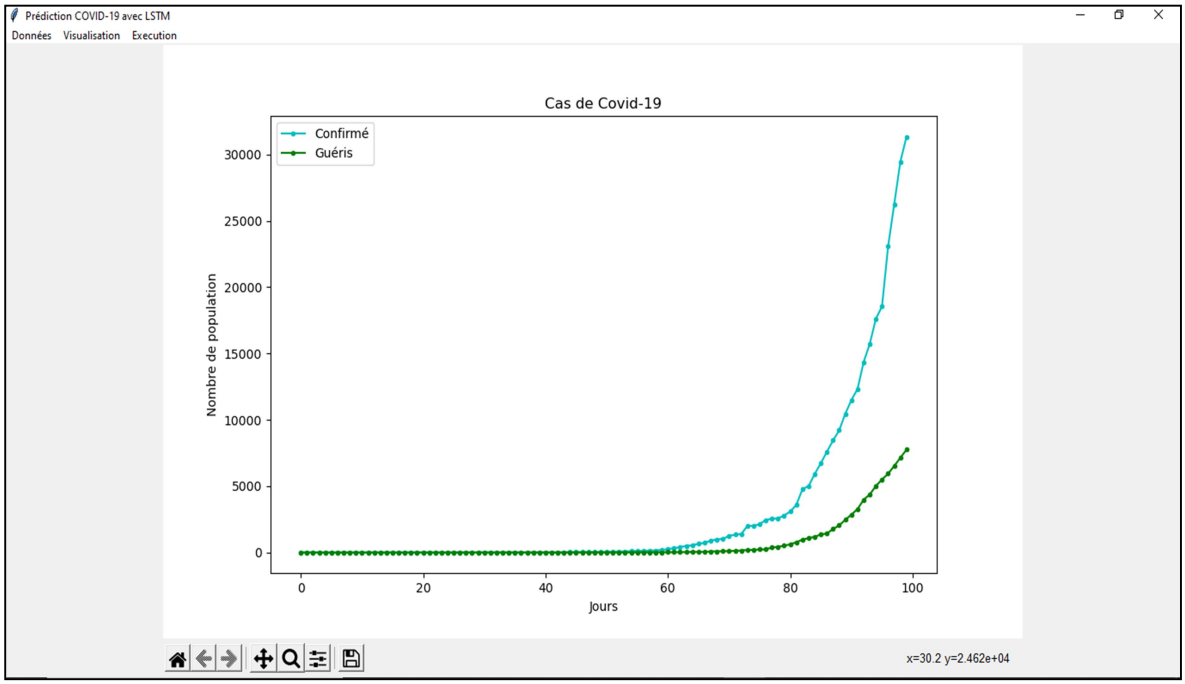


Figure 24: Interface de visualisation graphique des cas confirmés et guéris (tous).

Exécution

Pour cette partie lorsque l'utilisateur appuie sur **exécution→paramètres initiaux** tels que époques (le nombre de fois que l'algorithme déroule l'apprentissage sur l'ensemble de donnée), taux de validation(le taux de donnée distinct de l'ensemble d'apprentissage utilisé pour valider les performances de notre modèle), taille du lot(le nombre d'échantillons qui seront propagés sur le réseau) **→confirmé ou guéris**. Ces paramètres ont été initialisés par les valeurs telles que décrites dans le tableau2. L'application affiche un graphe qui représente les cas réels (ligne bleu) vs les cas prédits (ligne rouge) du type choisit après l'entrainement d'algorithme LSTM où l'axe verticale représente le nombre des cas et l'axe horizontal représente les jours (figure25).

Epoques	Taux de validation	Taille de lot
100	10%	5

Tableau 2: les valeurs des paramètres initiaux.

La figure ci-dessous montre que la ligne rouge (cas prédits) et la ligne bleu (cas réels) sont quasi-correspondantes ce qui implique que le choix des paramètres et le modèle sont fiables avec une précision optimale (figure25).

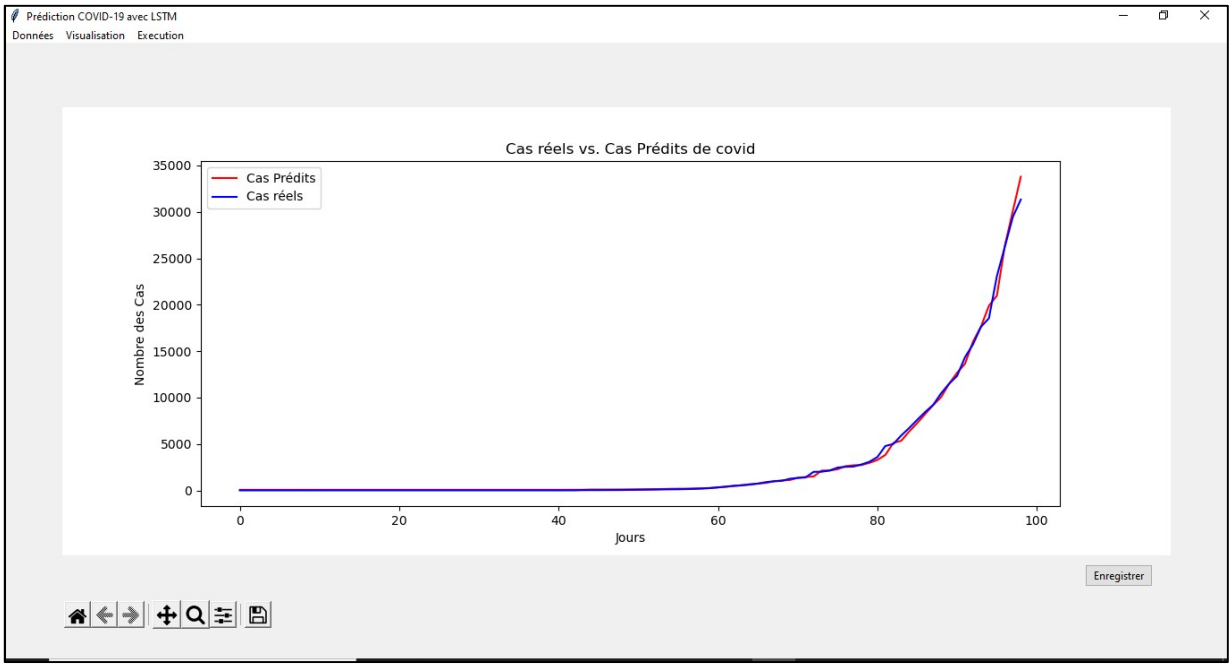


Figure 25: Graphe de prédiction avec paramètres initiaux.

L'application donne une option d'enregistrement des valeurs de la prédiction sous forme d'un fichier ayant comme extension « .csv » en appuyant sur **Enregistrer** dans un répertoire choisit comme le montre la figure 26.

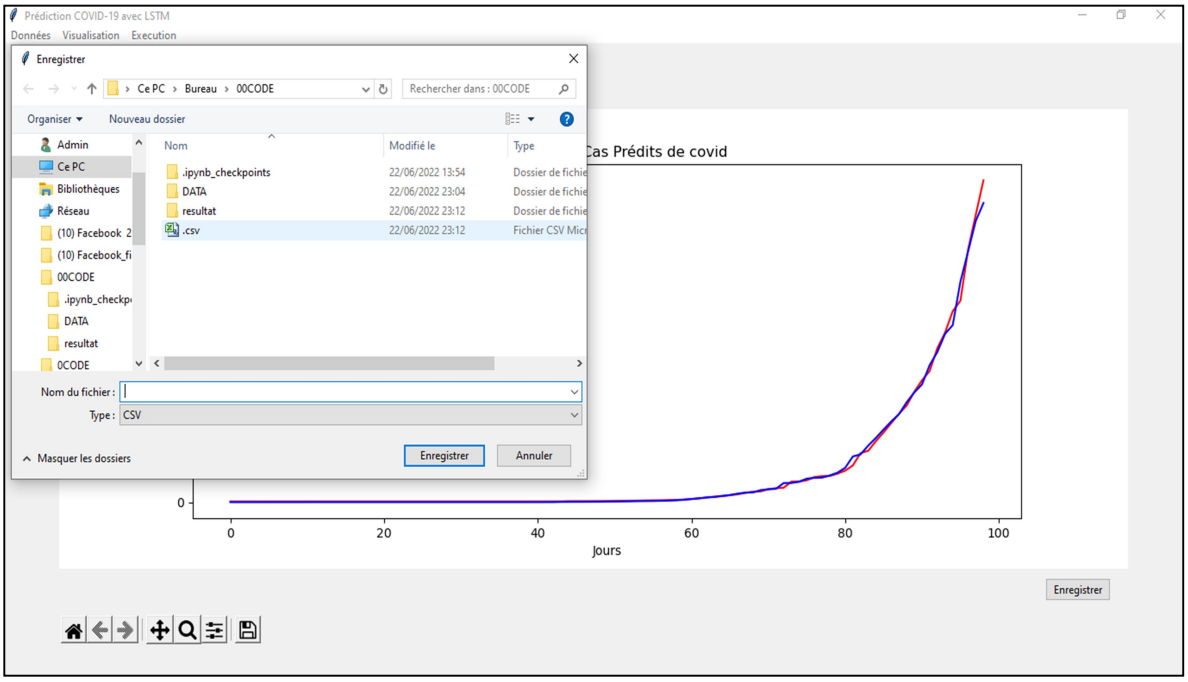


Figure 26: enregistrement avec paramètres initiaux.

Dans le cas d'une mauvaise coordination entre les résultats et les données réelles, l'application donne un autre choix à l'utilisateur pour changer les paramètres pour la prédiction en appuyant sur **exécution→changement de paramètres**. Une fois il introduit les trois paramètres en appuyant sur **exécuter**, le LSTM s'entraine et un graphe de prédiction s'affiche. Par exemple, nous avons déroulé pour la même base de données précédentes, d'autres valeurs telles que : taille de lot = 2, époques = 80, taux de validation = 0.5 (figure 27).Comme le montre la figure 27, la ligne des cas prédits se dissocie de la ligne des cas réels à partir du point 60 jours, ce qui valide le choixdes valeurs de paramètres cités dans la tableau2.

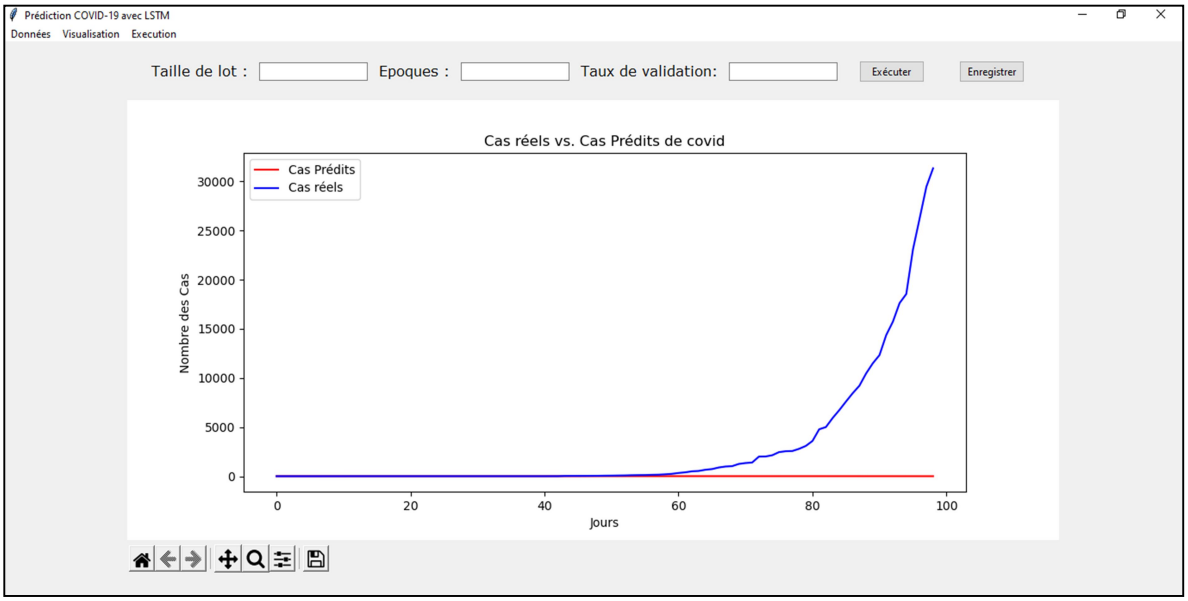


Figure 27: Graphe de prédiction avec paramètres changés.

Dans le cas d'enregistrement des valeurs avec les nouveaux paramètres, l'application enregistre les données dans un fichier csv ainsi que les paramètres avec l'emplacement du fichier enregistré dans un tableau (Figure 28).

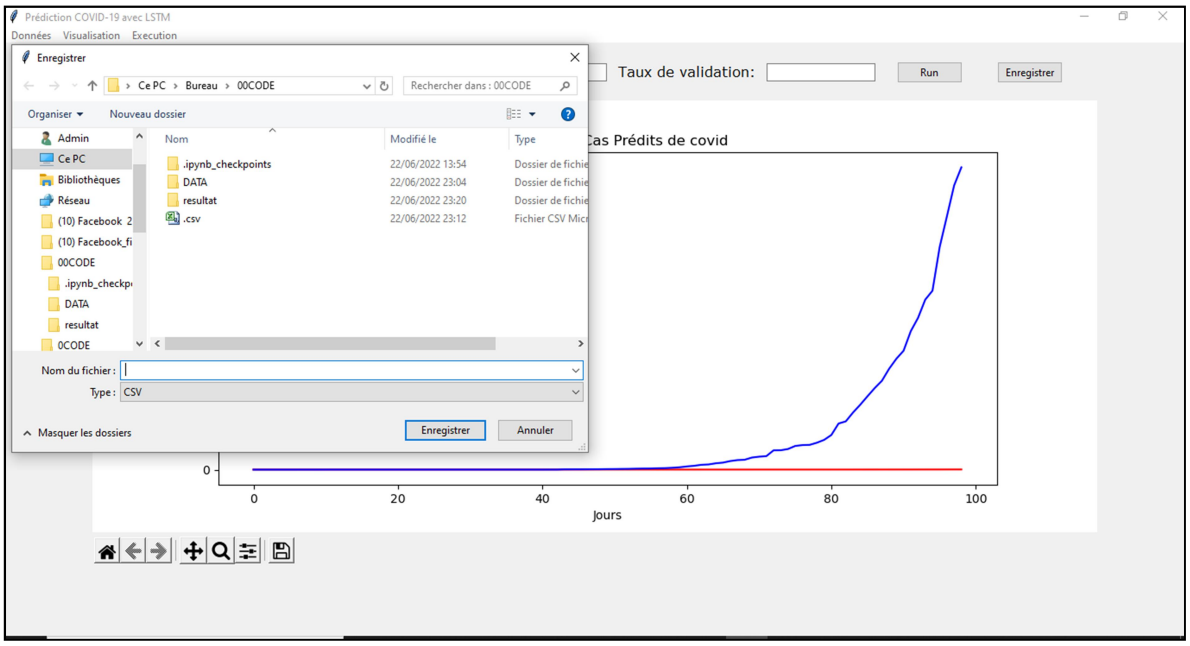
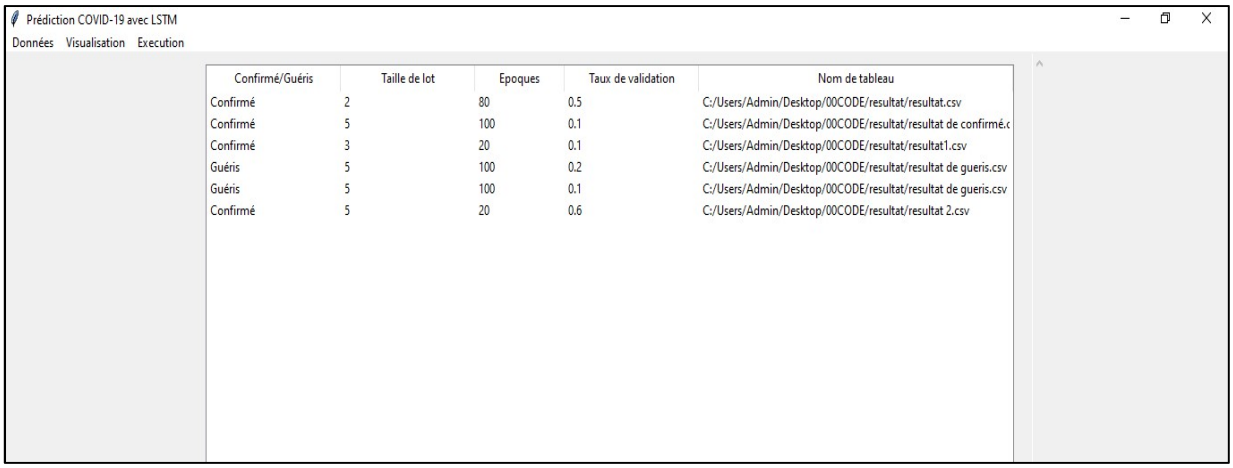


Figure 28: Enregistrement avec paramètres changés.

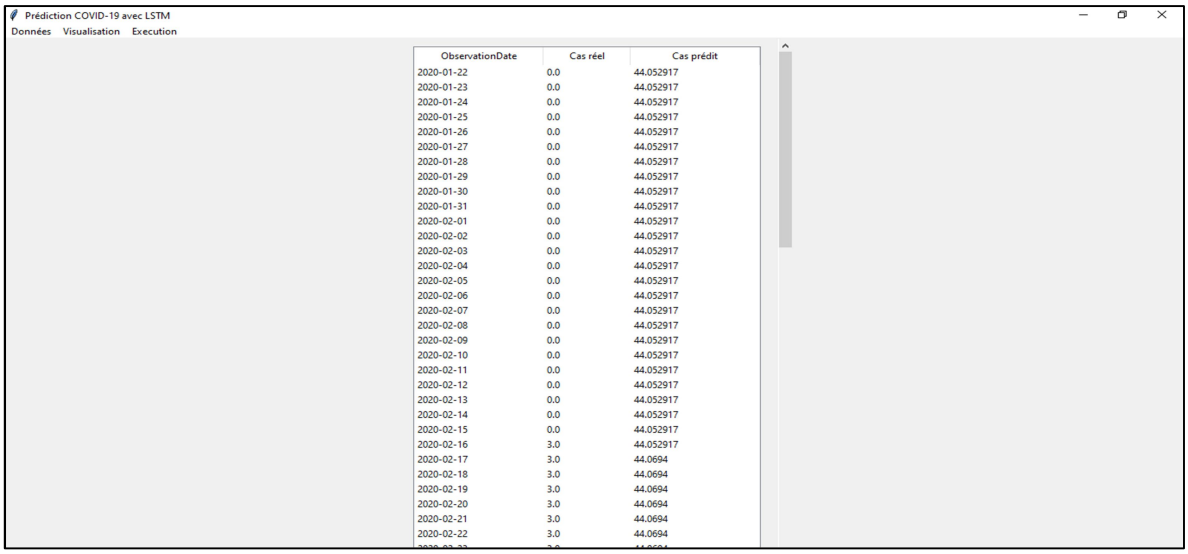
Après l'enregistrement, l'utilisateur peut consulter le tableau des paramètres en appuyant sur **exécution→consulter →paramètres** pour voir les paramètres utilisés et l'emplacement du résultat (figure 29)**.**



Confirmé/Guéris	Taille de lot	Epoques	Taux de validation	Nom de tableau
Confirmé	2	80	0.5	C:/Users/Admin/Desktop/00CODE/resultat/resultat.csv
Confirmé	5	100	0.1	C:/Users/Admin/Desktop/00CODE/resultat/resultat de confirmé.c
Confirmé	3	20	0.1	C:/Users/Admin/Desktop/00CODE/resultat/resultat1.csv
Guéris	5	100	0.2	C:/Users/Admin/Desktop/00CODE/resultat/resultat de gueris.csv
Guéris	5	100	0.1	C:/Users/Admin/Desktop/00CODE/resultat/resultat de gueris.csv
Confirmé	5	20	0.6	C:/Users/Admin/Desktop/00CODE/resultat/resultat 2.csv

Figure 29: Consultation des paramètres.

Au cas où l'utilisateur veut consulter les résultats en tableau, il appuie sur **exécution→consulter→résultat → tableau** pour afficher le résultat du fichier sous sélection (figure 30)**.**



ObservationDate	Cas réel	Cas prédit
2020-01-22	0.0	44.052917
2020-01-23	0.0	44.052917
2020-01-24	0.0	44.052917
2020-01-25	0.0	44.052917
2020-01-26	0.0	44.052917
2020-01-27	0.0	44.052917
2020-01-28	0.0	44.052917
2020-01-29	0.0	44.052917
2020-01-30	0.0	44.052917
2020-01-31	0.0	44.052917
2020-02-01	0.0	44.052917
2020-02-02	0.0	44.052917
2020-02-03	0.0	44.052917
2020-02-04	0.0	44.052917
2020-02-05	0.0	44.052917
2020-02-06	0.0	44.052917
2020-02-07	0.0	44.052917
2020-02-08	0.0	44.052917
2020-02-09	0.0	44.052917
2020-02-10	0.0	44.052917
2020-02-11	0.0	44.052917
2020-02-12	0.0	44.052917
2020-02-13	0.0	44.052917
2020-02-14	0.0	44.052917
2020-02-15	0.0	44.052917
2020-02-16	3.0	44.052917
2020-02-17	3.0	44.0694
2020-02-18	3.0	44.0694
2020-02-19	3.0	44.0694
2020-02-20	3.0	44.0694
2020-02-21	3.0	44.0694
2020-02-22	3.0	44.0694

Figure 30: Consultation des résultats en tableau.

Ou il peut consulter les résultats en graphe, en cliquant sur **exécution→consulter→résultat →graphe** pour afficher le résultat du fichier sous sélection (figure 31)**.**

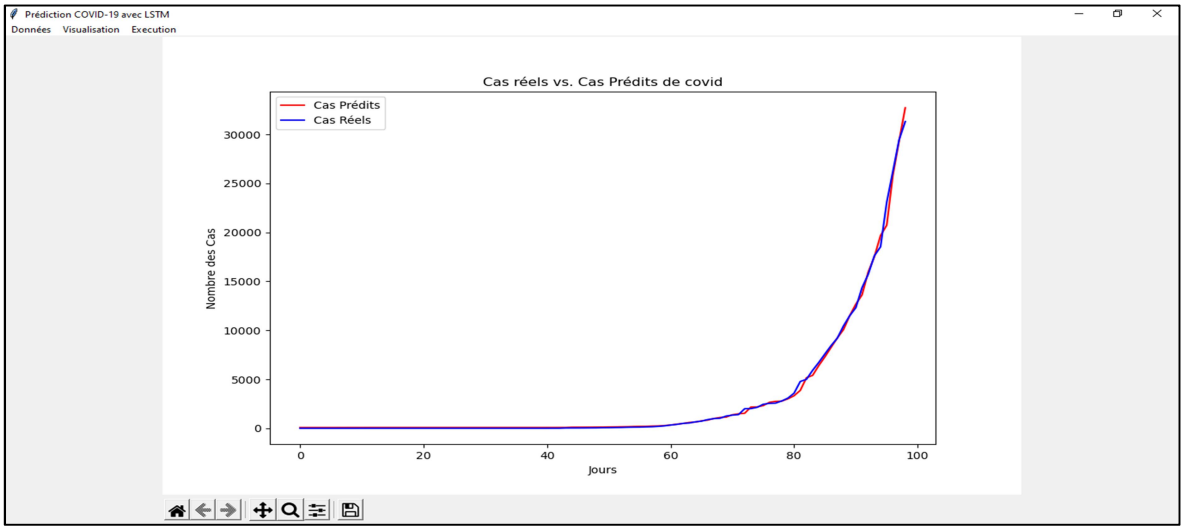


Figure 31: Consultation des résultats en graphe.

Nous avons déroulé une autre base de données qui est du pays Thaïlande (les données se trouvent ici : https://drive.google.com/uc?id=1edigloRtZGaShet2QW09I8vY1D4IUzQx) et les résultats sont montrés dans les figures (32 - 33).

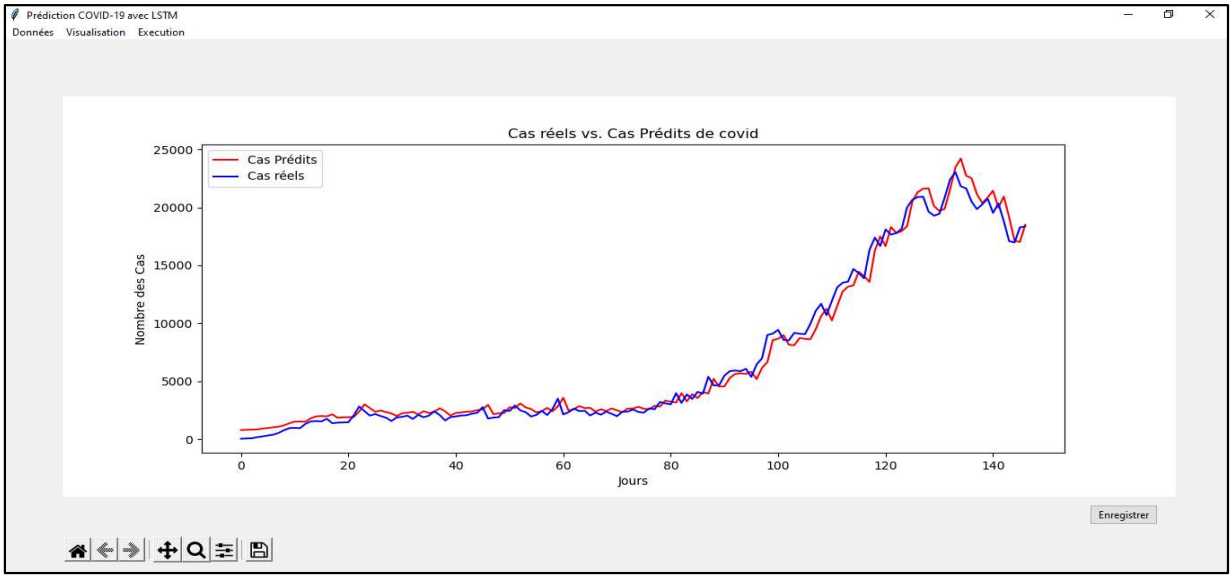


Figure 32 : Graphe de prédiction avec paramètres initiaux

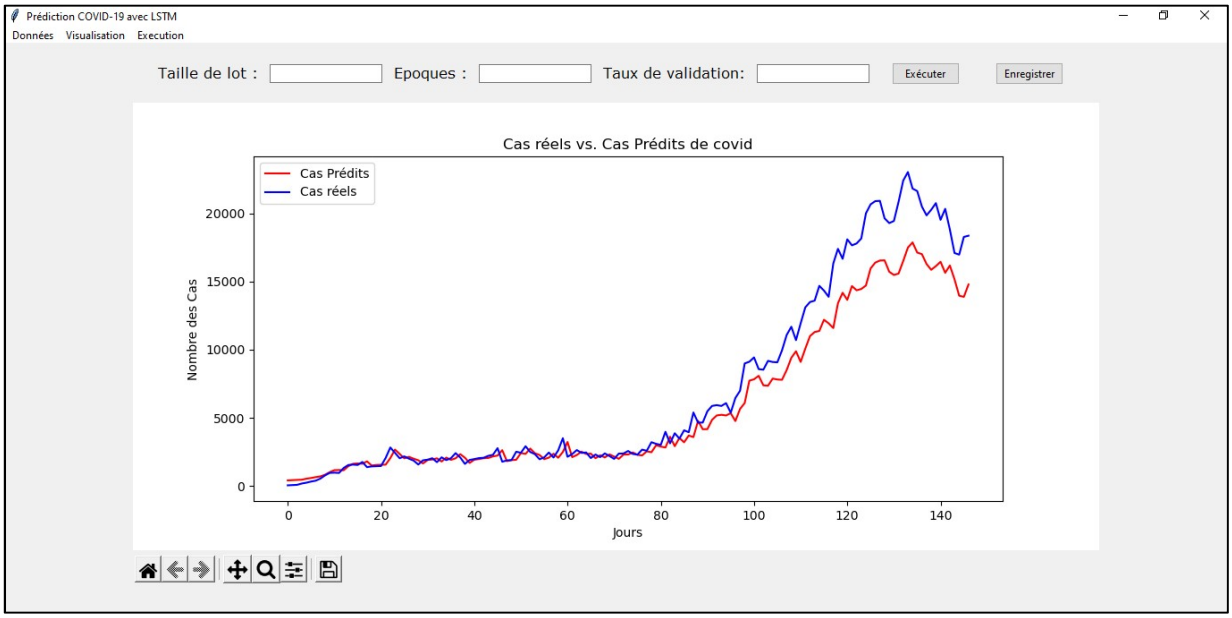


Figure 33 : Graphe de prédiction avec paramètres changés (taille de lot =2, époque=80, taux de validation =0.5)

3.5.3. Analyse

Ce que nous avons constaté durant cette expérience (Prédiction avec LSTM) après plusieurs essais que les paramètres qui influencent un changement dans le graphe de prédiction sont : taille de lot (batchsize accélère l'entrainement du réseau), époques (epochs influence sur la stabilité du réseau), taux de validation (validation split influence sur l'ensemble de donné choisi pour l'entrainement et la validation). Les valeurs des paramètres pour obtenir une prévision idéale sont : taille du lot=5, puisque elle améliore les performances du réseau et minimise la taille du mémoire, époques=100 c'est la valeur pour lequel le réseau est stabilisé. Après plusieurs essais de valeurs sur la division de l'ensemble de donnée pour l'entrainement et la validation nous avons déduit que taux de validation =0.1 donne un meilleur résultat que d'autre division.

Pour le Dropout = 0.5, ce paramètre est associe à l'architecture du LSTM, c'est-à-dire chaque neurone a 5 chance sur 10 d'être désactivé. A chaque époque, cette désactivation est appliquée aléatoirement. Pour mieux comprendre à chaque passe le modèle apprendra

avec une configuration de neurones différente, les neurones s'activant et se désactivant aléatoirement. La valeur de 0.5 approche les cas prédits des cas réels.

3.6. Conclusion

La première partie de ce dernier chapitre a été concrétisée par la présentation des différents outils utilisés pour la réalisation de notre application, après une définition de l'architecture de réalisation de l'application et l'architecture de notre modèle LSTM.

Dans la deuxième partie, nous avons abordé le coté de réalisation du projet avec des figures d'interfaces, ainsi qu'une analyse sur les résultats obtenus. Ce que nous avons obtenu à partir des résultats que le modèle LSTM fournit des meilleurs résultats de prédiction et avec une grande précision.

Conclusion générale

Selon l'Organisation mondiale de la santé (OMS), les maladies infectieuses sont responsables d'un quart à un tiers de tous les décès dans le monde. De plus, il est crucial de comprendre comment une maladie se propage dans la société moderne et d'identifier les foyers épidémiques. En effet, face à une telle menace, la société doit se préparer à l'avance pour réagir rapidement et efficacement si une telle épidémie se déclare. Ces derniers temps, la pandémie de la covid19 représente une urgence mondiale de santé publique à cause de transmission rapide. Dans le contexte de cette épidémie les épidémiologistes développent, testent et ajustent des modèles pour simuler la propagation de cette maladie infectieuse. La modélisation épidémiologique est largement utilisée dans l'analyse qualitative et quantitative de la propagation et du contrôle des maladies infectieuses afin de réduire la transmission de ce virus.

Selon la recherche bibliographique que nous avons effectuée, nous avons opté que les techniques de l'intelligence artificielle pour la modélisation de la covid19 se sont montrées efficaces dans plusieurs aspects de la lutte contre cette épidémie, car elles aident à comprendre le virus et accélérer la recherche ainsi que ralentir la propagation du virus.

En effet, notre recherche nous a mené à sélectionner l'apprentissage en profondeur comme choix pour modéliser l'épidémie de la Covid19, car les résultats de cette étude nous ont montré que l'apprentissage en profondeur peut détecter avec précision l'évolution de cette épidémie. Après une grande étude sur le domaine de l'apprentissage en profondeur dans la modélisation épidémiologique, nous avons pris le modèle LSTM comme choix pour la prédiction des futurs cas liées à cette épidémie.

La réalisation d'un tel projet nous a permis d'approfondir nos connaissances acquises tout le long de notre formation, et de maitriser de nouveaux langages et outils de développement. Malgré les difficultés que nous avons rencontrées pendant ce projet. Ce travail nous a donné l'opportunité de toucher une partie de divers aspects du métier de développeur et du concepteur.

Pour améliorer notre application, nous proposons comme perspectives d'automatiser la reconstruction du modèle LSTM en introduisant automatiquement le nombre de couches.

Annexe

Exposé : Un individu est dit exposé lorsqu'il est atteint d'une maladie infectieuse et qu'il n'est pas contagieux.

Infecté : Un individu qui a été atteint par la maladie et qui est contagieux.

Sensible : Une personne qui n'a pas encore été infectée et qui est exposée à la maladie est dite sensible.

En quarantaine : Une personne qui est infectée et en quarantaine pendant une certaine période

Guéri : Une personne qui a été infectée mais qui est maintenant guérie et en état de rétablissement.

Covid19 : Une maladie infectieuse causée par un coronavirus nouvellement découvert.

Épidémie : Apparition ou propagation rapide d'une maladie infectieuse, généralement par contagion, qui touche un grand nombre de personnes à la fois. Ce nombre dépasse le nombre de cas normalement attendus.

Pandémie : Une pandémie survient lorsqu'une grande partie de la population mondiale est infectée par un virus auquel le système immunitaire n'a pas encore développé une réponse adéquate.

Virus : Une particule microscopique infectieuse qui ne peut se répliquer qu'en pénétrant dans une cellule et en utilisant sa machinerie cellulaire

Intelligence artificielle : Consiste à mettre en œuvre un certain nombre de techniques visant à permettre aux machines d'imiter une forme d'intelligence réelle.

Bibliographie

[1] C. Dubé, G. Garner, M. Stevenson, R. Sanson et al : the use of epidemiological models for animal disease management,2007,pp 13-23.

[2] La modélisation au temps du covid 19 : https://uclouvain.be/fr/decouvrir/la-modelisation-au-temps-du-covid-19.html.

[3] Samuel Alizon : pour la SCIENCE : Modéliser les épidémies Ce que le Covid-19 a changé, 2021.

[4] Garner, M.G., Dubé, C., Stevenson, M.A., Sanson, R.L., Estrada, C., Griffin, J. :Evaluating alternative approaches to managing animal disease outbreaks. The role of modelling in policy formulation. Vet. Ital., 43 (2), 2007, pp 285-298.

[5] Garner, M.G., Beckett, S.D.: Modelling the spread of foot-and-mouth disease in Australia, AustVet. J., 83 (12), 2005, pp 758-766.

[6] Keeling, M.J., Eames, K.T.D: Networks and epidemic models, J. R. Soc. Interface, 2, 2005, pp 295-307.

[7] Taylor, N.: Review of the use of models in informingdisease control policydevelopment and adjustment. A report for DEFRA, Veterinary Epidemiology and Economics Research Unit (VEERU), School of Agriculture, Policy and Development, The University of Reading, UK, May 2003.

[8] Kitching, R.P., Thrusfield, M.V., Taylor, N.M : Use and abuse of mathematical models: an illustration from the 2001 foot and mouth disease epidemic in the United Kingdom, Rev. sci. tech. Off. int. Epiz., 25 (1), 2006, pp 293-311.

[9] James Holland Jones: Notes on R0, Department of Anthropological Sciences, Stanford University, California, May 2007.

[10] Lin Zhang, Xuyuan Wang, Juan Yang et al: Characterizing COVID-19 Transmission: Incubation Period, Reproduction Rate, and Multiple-GenerationSpreadin, Aristides (Aris) Moustakas, Natural History Museum of Crete, 8, 2021.

[11] Peter, H., thrall, Arjene Pierre, Marcy, K., uyenoyama : The American Naturalist : Frequency-Dependent Disease Transmission and the Dynamics of the Silene-Ustilago Host-Pathogen System, The University of Chicago,145(1), 1995, pp. 43-62.

[12] William Ogilvy Kermack and A. G. McKendrick : A contribution to the mathematicaltheory of epidemics, Royal Society, 115, 1927 pp. 700– 721.

[13] E. P. Gauthier Sallet : Mathematical Epidemiology: SEIR model, Institut Elie Cartan, UMR CNRS 7502, Université de Lorraine, 2018, pp. 41-43.

[14] Yongzhen Pei, Shaoying Liu, Shujing Gao, Shuping Li, Changguo Li : A delayed SEIQR epidemic model with pulse vaccination and the quarantinemeasure, Computers &Mathematicswith Applications, 58(1), July 2009, pp. 135-145.

[15] Higazy, M., Maryam Ahmed Alyami: New Caputo-Fabrizio fractionalorder SEIASqEqHR model for COVID-19 epidemic transmission with genetic algorithm based control strategy , behalf of Faculty of Engineering, Alexandria University 56(6),2020, pp 4719-4736.

[16] Fernando Concatto , Wellington Zunino, Luigi A. Giancoli, Raphaël Santiago, Luís C. Lamb :Geneticalgorithm for epidemic mitigation by removing relationships ,Proceedings of the Genetic and Evolutionary Computation Conference,2017,pp 761–768 ,

[17] Ganesh Kumar, M. , Soman, K. P., Gopala krishnan, E. A., Vijay Krishna Menon, Sowmya V.: Prediction of number of cases expected and estimation of the final size of coronavirus epidemic in India using the logistic model and genetic algorithm Ithaca, New York, Cornell University, 2020.

[18] Mingxing Wu, Liya Wang, Ming Li, Huijun Long: An approach based on the SIR epidemic model and a genetic algorithm for optimizing product feature combinations in feature fatigue analysis, Journal, Journal of Intelligent Manufacturing, 26, 2015, pp 199–209.

[19] Zandkarimi, M., Shafiei, M., Hadizadeh, F., Darbandi, M. A., Tabrizian, K.: Prediction of pharmacokinetic parameters using a genetic algorithm combined with an artificial neural network for a series of alkaloid drugs, SciPharm 2014, 82(1), 2013, pp 53-70.

[20] TAN, C., Chen, H., Wu, T., Xia, C. : Modeling the Relationship Between Cervical Cancer Mortality and Trace Elements Based on Genetic Algorithm–Partial Least Squares and Support Vector Machines Biological Trace Element Research 140, 2011, pp 24–34.

[21] Aleta , A., Martín-Corral, D., Pastorey Piontti, A., Ajelli, M., Litvinova, M., Chinazzi, M., Dean, N. E., Halloran, M. E., Longini Jr, I. M., Merler, S., Pentland, A., Vespignani, A., Moro, E., Moreno, Y., Behav, N. H.: Modeling the impact of social distancing, testing, contact tracing and house hold quarantine on second-wave scenarios of the COVID-19 epidemic. medRxiv,4, 2020, pp 964.

[22] Ferguson, N. M., Laydon, D., Nedjati-Gilani, G. and al: Impact of non-pharmaceutical interventions (NPIs) to reduce COVID-19 mortality and healthcare demand, Report of the Imperial College London, Faculty of Medicine, Imperial College London, London, 2020.

[23] Gressman, P.T., PECK, J.R.: Simulating COVID-19 in a university environment, Mathematical Biosciences, 328, 2020,pp 108-436.

[24] Hinch , R., Probert, W.J.M., Nurtay, A., Kendall, M. and al :OpenABM-Covid19 - an agent-based model for non-pharmaceutical interventions against COVID-19 including contact tracing , medRxiv ,2 , 2020.

[25] Khaled, M., Khalil, Abdel-Aziz, M., Taymour, T., Nazmy, Abdel-Badeeh, M., Salem : An Agent-Based Modeling for Pandemic Influenza in Egypt, Intelligent Systems Reference Library, 33, 2010, pp 205–218.

[26] PEREZ, L., Dragicevic, S: An agent-based approach for modeling dynamics of contagious disease spread, International Journal of Health Geographics 8(1), 2009.

[27] Piotr Podolski , Hung Son Nguyen: cellular Automata in Covid-19 prediction, Procedia Computer Science, 192, 2021, Pages 3370-3379.

[28] P.H.T.Schimit : A model based on cellular automata to estimate the social isolation impact on COVID-19 spreading in Brazil, Computer Methods and Programs in Biomedicine, 200, 2021.

[29] P.K.Jithesh :A model based on cellular automata for investigating the impact of lockdown, migration and vaccination on COVID-19 dynamics , Computer Methods and Programs in Biomedicine, 211, 2021.

[30] Sayantari Ghosh, Saumik Bhattacharya: Computational Model on COVID-19 Pandemic Using Probabilistic Cellular Automata, SN Computer Science, 2, 2021.

[31] SayantariGhosh, SaumikBhattacharya: A data-driven understanding of COVID-19 dynamics using sequential genetic algorithm based probabilistic cellular automata, Applied Soft Computing, 96, 2020.

[32] Hoya White, S., Martín del Rey, A., Rodríguez Sánchez, G. : Modeling epidemics using cellular automata, 2006,Applied Mathematics and Computation,186, 2007, pp 193-202.

[33] Fuentes, M.A., Kuperman, M.N.: Cellular automata and epidemiological models with spatial dependence, Physica A: Statistical Mechanics and its Applications, 267, 1999, pp 471-486.

[34] Khan, S.D., Alarabi, L., Basalamah, S.: Toward smart lockdown: A novel approach for COVID-19 hotspots prediction using a deephybrid neural network Computers, The first computers , 9 (4), 2020, p 99.

[35] Farooq, J., Bazaz, M.A.: A Deep Learning algorithm for modeling and forecasting of COVID-19 in five worstaffected states of India, Alex. Eng. J., 60 (1), 2021, pp. 587-596.

[36] Jin, B., Ji, J., Yang, W., Yao, Z., Huang, D., Xu, C.: Analysis on the spatio-temporal characteristics of COVID-19 in mainland China, Process Safety and Environmental Protection, 152, 2021, Pages 291-303.

[37] Aishwarya, T., Kumar, V.R.: Machine learning and Deep Learning approaches to analyze and detect COVID-19: a review, SN Comput. Sci., 2 (3), 2021, pp. 1-9.

[38] Jayanthi, Devaraj , Rajvikram Madurai Elavarasan and al: Forecasting of COVID-19 cases using Deep Learning models: Is itreliable and practically significant?, Results in Physics, 21, 2021.

[39] Yuexin Wu, Yiming Yang, Hiroshi Nishiura, Masaya Saitoh: Deep Learning for Epidemiological Predictions, SIGIR '18: The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval , 26, June 2018, pp 1085–1088.

[40] Futura Santé, https://www.futura-sciences.com/sante/definitions/coronavirus-covid-19-18585 .

[41] WHO (World Health Organization) : Chronologie de l'action de l'OMS face à la COVID-19, https://www.who.int/fr/news/item/29-06-2020-covidtimeline

[42] Arbel, Y., Fialkoff, C., Kerner, A., Kerner, M.: Do population density, socio-economic ranking and Gini index of cities influence infection rates from coronavirus?, The Annals of Regional Science volume 68, 2022, pp181–206.

[43] Ehlert, A.: The socioeconomic determinants of COVID-19: A spatial analysis of German county level data, Socio-Economic Planning Sciences,78, 2021.

[44] Wong, D.W.S., Li, Y. : Spreading of COVID-19: Density matters, PLoS ONE ,15,2020.

[45] Jean Paul-Sardon : De la longue histoire des épidémies au Covid-19, Les Analyses de Population & Avenir, 8, 2020, pp 1-18.

[46] Boughaba, M., Boukhris, B.: L`apprentissage profond (Deep Learning) pour la classification et la recherche d'images par le contenu, Département d'Informatique et des Technologies de l'information, université de kasdi merbah, ouargla, Alger, 2018.

[47] blogs : Intelligence Artificielle et Social Media Listening : Les exemples concrets,https://blog.digimind.com/fr/insight-driven-marketing/intelligence-artificielle-social-media-listening-applications-concretes.

[48] Simplilearn : Top 10 Deep Learning Algorithms You Should Know in 2022, https://www.simplilearn.com/tutorials/deep-learning-tutorial/deep-learning-algorithm.

[49] Intelligence artificielle, https://intelligence-artificielle.com/top-algorithme-deep- earning/.

[50] Zhifang Liao, Peng Lan, Xiaoping Fan, Benjamin Kelly, Aidan Innes: SIRVD-DL: A COVID-19 Deep Learning prediction model based on time-dependent SIRVD, Computers in Biology and Medicine, 138, 2021.

[51] Junaid Farooq, Mohammad Abid Bazaz : A novel adaptive Deep Learning model of Covid-19 with focus on mortality reduction strategies,2020, Chaos, Solitons & Fractals,138, 2020.

[52] Hyeontae Jo, Hwijae Son, Hyung Ju Hwang, Se Young Jung: Analysis of the spread of COVID-19 in South Korea using the SIR model with time-dependent parameters and Deep Learning, medRxiv, 2020.

[53] Jayanthi Devaraj, Rajvikram Madurai Elavarasan and al: Forecasting of COVID-19 cases using Deep Learning models: Is itreliable and practically significant?, Results in Physics, 21, 2021.

[54] Démystifier le Machine Learning, Partie 2 : les Réseaux de Neurones artificiels, https://www.juripredis.com/fr/blog/id-19-demystifier-le-machine-learning-partie-2-les-reseaux-de-neurones-artificiels.

[55] Claude Touzet: Les Réseaux de Neurones Artificiels: Introduction au Connexionnisme, Aix Marseille Université, Marseille, France , 1992.

[56] Deep Learning : les réseaux de neurones récurrents, www.datavalue-consulting.com, https://datavalue-consulting.com/deep-learning-reseaux-neurones-recurrents-rnn.

[57] Illustrated Guide to LSTM's and GRU's: A step by stepexplanation ofMichael PhiTowards Data Science, https://towardsdatascience.com.

[58] DataBird:https, //www.data-bird.co/python/langage-python.

[59] Anaconda, https://www.anaconda.com.

[60] JupyterNotebook, https://jupyter.org.

[61] JDN, https://www.journaldunet.fr/web-tech/guide-de-l-intelligence-artificielle/1501863-keras-bibliotheque-de-deep-learning/.

[62] Tkinter, https://docs.python.org/3/library/tkinter.html.

[63] Dilip Kumar Bagal, Arati Rath, Abhishek Barua, Dulu Patnaik: Estimating the parameters of susceptible-infected-recovered model of COVID-19 cases in India during lock down periods, Chaos, Solitons & Fractals, 140, 2020.