



جامعة بجاية
Tasdawit n Bgayet
Université de Béjaïa

UNIVERSITÉ ABDERRAHMANE MIRA DE BEJAÏA
FACULTÉ DES SCIENCES EXACTES
DÉPARTEMENT D'INFORMATIQUE

MÉMOIRE DE MASTER RECHERCHE

Option : Intelligence Artificielle

T H È M E

Proposition d'un Système Multi-Agents Basé sur le
Deep Learning pour la Prédiction du Risque de
Développement d'une Rétinopathie Diabétique

Réalisé par M^{elle} :

GUISSI Lydia

Devant le jury composé de :

Premier examinateur : M^r AMROUN Kamal
Deuxième examinateur : M^r BOUCHEBBAH Fatah
Encadrante : M^{me} BOUKREDERA Djamilia
Co-encadrante : M^{me} AIT KACI AZZOU Samira

Promotion 2021-2022

Remerciements

Avant tout je remercie Dieu « ALLAH » le tout puissant de m'avoir accordé la force, le courage et la patience dont j'avais besoin pour mener à bien ce travail.

La réalisation d'un mémoire est la somme d'un travail collectif où l'apport de chacun aussi anodin soit-il, est toujours indispensable et précieux, cela m'a permis de me rendre compte que rien n'est impossible et que l'union fait la force !

Je remercie d'abord mes encadrantes Mme. BOUKREDERA DJAMILA et Mme. AIT KACI AZZOU SAMIRA qui m'ont permis de toucher à un travail intéressant et important pour une grande partie de la population.

Mes vifs remerciements vont également aux membres du jury : Mr. AMROUN KAMAL et Mr. BOUCHEBBAH FATAH pour l'intérêt qu'ils ont porté à mon travail en acceptant d'examiner ce dernier et de l'enrichir par leurs propositions.

Enfin, je tiens également à remercier toutes les personnes ayant participé de près ou de loin à la réalisation de ce travail, dont mes parents, mes racines, qui m'ont transmis tant de choses dont la persévérance, l'honnêteté et l'une de mes plus grande richesse l'amour du travail.

Dédicace

*À mon cher père et à ma chère mère, qui à mes yeux sont bien plus que de simples parents.
À ces deux êtres inestimables, source d'inspiration et de motivation qui ont toujours été là
pour m'encourager et pour faire de moi la personne que je suis aujourd'hui.*

*Aucune dédicace ne saurait exprimer mon respect, mon amour et mon admiration éternelle
ainsi que ma considération pour les innombrables sacrifices dont vous avez fait preuve tout au
long de ces années.*

Puisse dieu, le tout puissant, vous accorder bonheur, joie, bonne santé et longue vie.

*À mes chères sœurs : Celia, Inès et Hanane. Avoir des sœurs comme vous est une chance
inouïe, merci pour votre soutien. J'aimerais vous exprimer mon attachement et mon affection,
qu'Allah vous bénisse et vous accorde joie, réussite et bonheur dans vos vies.*

*À toutes les personnes qui ont participé à ce projet de près ou de loin . Merci pour votre
dévouement et votre patience afin de mener à bien ce projet.*

À tous mes amis.

Aux souvenirs que nous garderons en mémoire.

Aux moments passés ensemble.

Je vous souhaite bonheur et réussite.

Lydia

Table des matières

Table des matieres	i
Table des figures	v
Liste des acronymes et des abréviations	vii
Introduction Générale	1
1 Généralités sur la rétinopathie diabétique	3
Introduction	3
1.1 L’anatomie de l’oeil	3
1.2 La cause de la Rétinopathie Diabétique	4
1.3 Les stades de la rétinopathie diabétique	4
1.3.1 Rétinopathie diabétique non proliférante minime (ou légère)	4
1.3.2 Rétinopathie diabétique non proliférante modérée	5
1.3.3 Rétinopathie diabétique non proliférante sévère	5
1.3.4 Rétinopathie diabétique proliférante	6
Conclusion	7
2 Application des techniques de l’IA dans le diagnostic médical	8
Introduction	8
2.1 Techniques de l’apprentissage automatique pour la classification et la prédiction de la rétinopathie diabétique	8
2.1.1 Machine Learning	9
2.1.2 Types d’algorithmes de Machine Learning	9
2.1.2.1 Apprentissage supervisé	9
2.1.2.2 Apprentissage non supervisé	9
2.1.2.3 Apprentissage par renforcement	9
2.1.3 Deep Learning	9
2.1.3.1 Réseaux neuronaux artificiels	10
2.1.3.2 Fonctions d’activation	10
2.1.3.3 Réseau neuronal convolutif (CNN)	12
2.1.4 Application de l’apprentissage automatique pour la classification de la rétinopathie diabétique	12
2.1.4.1 Réseaux neuronaux convolutifs pour la rétinopathie diabétique[1]	12

2.1.4.2	Classification d'images de rétinopathie diabétique à l'aide de modèles d'apprentissage profond[2]	13
2.1.4.3	Classification des stades de la rétinopathie diabétique à l'aide de réseaux de neurones convolutifs [3]	13
2.1.4.4	Filtrage automatique des images du fond d'œil à l'aide d'une combinaison de réseaux neuronaux convolutionnels et du hand-crafted features[4]	14
2.1.4.5	Évaluation diagnostique des algorithmes d'apprentissage profond pour le dépistage de la rétinopathie diabétique[5]	14
2.1.4.6	Architecture Alexnet modifiée pour la classification des images de rétinopathie diabétique[6]	14
2.1.5	Application de l'apprentissage automatique pour la prédiction de la rétinopathie diabétique	15
2.1.5.1	Calculateurs de risque de rétinopathie dans la prédiction de la rétinopathie diabétique menaçant la vue dans le diabète de type 2 : Une sous-étude FIELD[7]	15
2.1.5.2	Évaluation individualisée du risque de rétinopathie diabétique et optimisation des intervalles de dépistage : une approche scientifique pour réduire les coûts des soins de santé[8]	15
2.1.5.3	Modèle prédictif et analyse du risque de rétinopathie diabétique à l'aide de l'apprentissage automatique : une étude de cohorte rétrospective en Chine[9]	16
2.1.5.4	Prévision de la rétinopathie diabétique et identification de caractéristiques biomédicales interprétables à l'aide d'algorithmes d'apprentissage automatique [10]	16
2.1.5.5	Prédiction du risque de rétinopathie diabétique à partir de l'examen du fond d'œil en utilisant l'apprentissage clairsemé : une étude transversale[11]	17
2.2	Approches multi-agents pour le diagnostic médical	17
2.2.1	Définitions	17
2.2.2	Avantages des systèmes multi-agents	18
2.2.3	Application des systèmes multi-agents dans le domaine du Healthcare	19
2.2.3.1	Agent d'apprentissage profond amélioré et robuste pour la détection préliminaire de la rétinopathie diabétique[12]	20
2.2.3.2	Détection d'Alzheimer basée sur des systèmes multi-agents : Un environnement intelligent de traitement d'images[13]	20
2.2.3.3	Contrôle optimisé pour la segmentation d'images médicales : accords améliorés de systèmes multi-agents à l'aide de l'optimisation par essaims (ParticleSwarm Optimization)[14]	21
2.2.3.4	Un modèle d'optimisation multi-agents piloté par les bords pour la détection des maladies infectieuses[15]	21
2.2.3.5	Une approche multi-agents d'apprentissage par renforcement profond pour améliorer la segmentation des images tomographiques COVID-19[16]	21

2.2.3.6	Utilisation d'un système multi-agents pour la détection des microanévrismes dans les images de fond de l'œil[17]	22
2.2.3.7	Une approche multi-agents pour la prédiction personnalisée de l'hypertension artérielle[18]	22
	Conclusion	23
3	Un modèle multi-agents basé sur le Deep Learning pour la prédiction de la rétinopathie diabétique	24
	Introduction	24
3.1	Architecture du modèle multi-agents proposé	24
3.1.1	Rôles et architecture interne des agents	27
3.1.1.1	Agent utilisateur	27
3.1.1.2	Agent classification	27
3.1.1.3	Agent prédiction	28
3.1.2	Dynamique du système multi-agents proposé	29
	Conclusion	30
4	Expérimentation	31
	Introduction	31
4.1	outils et Librairies utilisés	31
4.1.1	Anaconda	31
4.1.1.1	Spyder	31
4.1.2	Python	31
4.1.2.1	SPADE	32
4.1.2.2	Matplotlib	32
4.1.2.3	Pandas	32
4.1.2.4	Numpy	32
4.1.2.5	Scikit-learn	32
4.1.2.6	Scikit-plot	32
4.1.2.7	Tensorflow	32
4.2	Définition de l'ensemble des données utilisé et description des variables	33
4.2.1	Définition de l'ensemble des données utilisé	33
4.2.2	Description des variables	34
4.3	Implémentation de l'agent Prédiction	35
4.3.1	Utilisation de la méthode Random Forest	35
4.3.1.1	Implémentation	35
4.3.1.2	Résultats obtenus avec Random Forest	37
4.3.1.3	Scénario d'exécution	39
4.3.2	Utilisation du Deep Learning	40
4.3.2.1	Architecture du réseau de neurones	40
4.3.2.2	Implémentation	41
4.3.2.3	Résultats obtenus avec l'approche Deep Learning	43
4.3.2.4	Scénario d'exécution	43
4.4	Discussion des résultats	45

Conclusion	45
Conclusion Générale	46
Bibliographie	47

Table des figures

1.1	Description de l'anatomie de l'oeil [19]	4
1.2	Rétinopathie diabétique non proliférante minime [20]	5
1.3	Rétinopathie diabétique non proliférante modérée [20]	5
1.4	Rétinopathie diabétique non proliférante sévère [20]	6
1.5	Rétinopathie diabétique proliférante [21]	6
2.1	Réseaux neuronaux artificiels	10
2.2	Fonction Hyperbolic tangent[22]	11
2.3	Fonction ReLU[22]	11
2.4	Fonction Sigmoid[22]	11
3.1	Architecture du modèle multi-agents proposé	26
3.2	Architecture interne de l'agent Classification	28
3.3	Architecture interne de l'agent prédiction	29
3.4	Diagramme d'interactions du système multi-agents	30
4.1	Aperçu du dataset adapté	33
4.2	Bibliothèques importées pour Random Forest	36
4.3	Importation du dataset et modifications	36
4.4	Aperçu après modification	36
4.5	Séparation des données en données testes et entraînement	37
4.6	Random Forest	37
4.7	Instruction pour l'affichage de l'accuracy de Random Forest	37
4.8	Instruction pour l'affichage du rapport de classification de Random Forest	38
4.9	Rapport de classification de Random Forest	38
4.10	Instruction pour l'affichage de la matrice de confusion de Random Forest	38
4.11	Matrice de confusion de Random Forest	38
4.12	Profils de trois patients pour tester le processus de prédictions	39
4.13	Instruction pour prédire le risque d'apparition de la rétinopathie diabétique chez le patient N°1	39
4.14	Résultat de la prédiction du risque d'apparition de la rétinopathie diabétique chez le patient N°1	39
4.15	Instruction pour prédire le risque d'apparition de la rétinopathie diabétique chez le patient N°2	40
4.16	Résultat de la prédiction du risque d'apparition de la rétinopathie diabétique chez le patient N°2	40

4.17	Instruction pour prédire le risque d'apparition de la rétinopathie diabétique chez le patient N°3	40
4.18	Résultat de la prédiction du risque d'apparition de la rétinopathie diabétique chez le patient N°3	40
4.19	Résumé du modèle	41
4.20	Bibliothèques importées pour le Deep Learning	41
4.21	Division du jeu de données en données de tests et données d'entraînement pour le Deep Learning	42
4.22	Instructions pour la création de notre réseau de neurones et sa compilation . . .	42
4.23	Application du EarlyStopping	43
4.24	Instructions pour obtenir l'accuracy du modèle de Deep Learning développé . .	43
4.25	Résultats obtenus avec le modèle de Deep Learning développé	43
4.26	Instruction pour prédire le risque d'apparition de la rétinopathie diabétique chez le patient N°1 avec Deep Learning	44
4.27	Résultat de la prédiction du risque d'apparition de la rétinopathie diabétique chez le patient N°1 en utilisant le Deep Learning	44
4.28	Instruction pour prédire le risque d'apparition de la rétinopathie diabétique chez le patient N°2 en utilisant le Deep Learning	44
4.29	Résultat de la prédiction du risque d'apparition de la rétinopathie diabétique chez le patient N°2 en utilisant le Deep Learning	44
4.30	Instruction pour prédire le risque d'apparition de la rétinopathie diabétique chez le patient N°3	44
4.31	Résultat de la prédiction du risque d'apparition de la rétinopathie diabétique chez le patient N°3 avec Deep Learning	45

Liste des acronymes et des abréviations

AMIR	Anomalies Microvasculaires Intra Rétiniennes
AUC	Area Under the Curve
BNN	Back-propagation Neural Network
CHU	Centre Hospitalier Universitaire
CNN	Convolutional Neural Network
DNN	Deep Neural Network
DR	Diabetic Retinopathy
FN	Faux Négatif
FP	Faux Positif
IA	Intelligence Artificielle
IRM	Imagerie par Résonance Magnétique
MAS	Multi-agent System
ML	Machine Learning
RD	Rétinopathie Diabétique
SMA	Système Multi-Agents
VN	Vrai Négatif
VP	Vrai Positif

Introduction Générale

La rétinopathie diabétique(RD) est une maladie fréquente qui provoque une perte de vision, chez les personnes atteintes de diabète touchant 50% des patients diabétiques de type 2. La détection précoce de cette maladie permet d'éviter les complications. Ainsi, il est primordial de pouvoir prédire le risque de développement d'une rétinopathie diabétique chez un patient atteint de diabète. En effet, la rétinopathie diabétique peut être évitée en contrôlant les facteurs de risque.

Durant les dernières décennies, des systèmes automatisés d'aide à la décision dans le domaine de la santé ont vu le jour afin d'aider les cliniciens d'établir de bons diagnostics. C'est dans ce contexte que les techniques de l'intelligence artificielle ont été très utilisées dans la santé et plus particulièrement en ophtalmologie.

Lorsque l'on parle d'IA et du domaine médical nous pensons à la médecine prédictive, notamment la prédiction d'une maladie et de son développement mais aussi l'aide à la décision pour établir un diagnostic médical. La plupart des techniques utilisées sont des techniques de l'apprentissage automatique. Les réseaux de neurones ont été les plus utilisés pour la détection et la classification de diverses pathologies.

Par ailleurs, ces techniques ont montré leurs limites quand il s'agit de systèmes dynamiques et complexes. Les systèmes multi-agents se sont alors avérés une solution à ce problème. En effet, Les systèmes multi-agents ont révélé leur potentiel d'améliorer les systèmes distribués, parallèles et concurrents de diverses manières.

Combinaison un système d'apprentissage automatique avec un système multi-agents permet de développer des modèles de prédiction intelligents et efficaces.

L'objectif principal de notre travail est de développer un système multi-agent basé sur les techniques du Machine Learning et du Deep Learning capable de détecter et de effectuer une prédiction personnalisée du risque de développement d'une rétinopathie diabétique.

Le modèle que nous proposons est composé de plusieurs agents communicants dont la structure interne est un réseau de neurones. Nous avons procédé à plusieurs expérimentations avec différents scénarios et les résultats obtenus sont très prometteurs. La suite de ce mémoire est structurée comme suit :

- **Chapitre 1 : Généralités sur la rétinopathie diabétique.** Ce chapitre va discuter de la rétinopathie diabétique en passant de l'anatomie de l'œil à la cause de la RD mais aussi aux stades de cette maladie.

- **Chapitre 2 : Application des techniques de l'IA dans le diagnostic médical.** Nous allons définir l'apprentissage automatique et tout ce qui lui est relatif ; nous allons aussi présenter un état de l'art de la classification et de la prédiction de la rétinopathie diabétique avec usage de machine Learning. Par la suite, nous présenterons les systèmes multi-agent, leurs avantages ainsi qu'un état de l'art des SMA dans le domaine du healthcare.
- **Chapitre 3 : Un modèle multi-agents basé sur le Deep Learning pour la prédiction de la rétinopathie diabétique.** Dans ce chapitre, nous allons présenter notre modèle de classification et de prédiction du risque de développement de la RD. La structure et la dynamique de notre modèle ont été bien détaillées.
- **Chapitre 4 : Expérimentation.** Dans ce chapitre, nous allons présenter les différents environnements et outils nécessaires au développement de notre modèle ainsi que les résultats d'expérimentation.
- **Conclusion Générale .** Une conclusion générale résumera l'essentiel de notre travail.

Chapitre 1

Généralités sur la rétinopathie diabétique

Introduction

Le diabète est une maladie chronique qui survient lorsque le pancréas n'est plus capable de produire de l'insuline, ou lorsque l'organisme ne peut pas utiliser correctement l'insuline qu'il produit [23]. La rétinopathie diabétique ou maladie oculaire diabétique est une complication redoutée du diabète [24]. Dans ce chapitre nous allons tout d'abord définir l'anatomie de l'œil, ensuite nous citerons la cause de la rétinopathie diabétique pour enfin présenter les stades de cette maladie.

1.1 L'anatomie de l'œil

Le globe oculaire est un organe sphérique situé dans l'orbite et est responsable de la fonction visuelle [25]. Il est constitué de trois enveloppes : la sclérotique (enveloppe externe), l'uvée (enveloppe intermédiaire) et la rétine (enveloppe interne) [25].

La rétine a pour fonction de traiter les informations visuelles par l'intermédiaire des photorécepteurs présents dans l'œil, appelés bâtonnets et cônes. Au centre de la rétine se trouve une zone ronde et sombre appelée Macula. Le point central de la macula, appelée fovéa, est chargé de fournir une vision nette. Comme tout autre tissu de l'organisme, le tissu rétinien est alimenté en sang par le système vasculaire du corps. Le disque optique, composé de la cupule optique, est une structure où le nerf optique quitte l'œil. C'est aussi le lieu où toutes les grandes artères et veines entrent et sortent de l'œil [26].

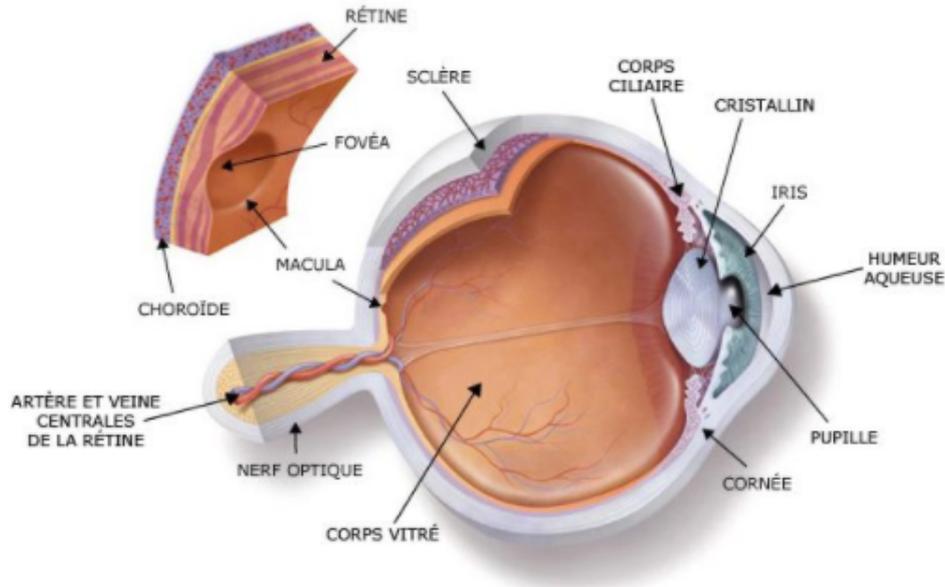


FIGURE 1.1 – Description de l’anatomie de l’oeil [19]

1.2 La cause de la Rétinopathie Diabétique

La rétine est une fine membrane de l’oeil traversée par plusieurs petits vaisseaux appelés capillaires. L’excès de sucre dans le sang comme dans le cas d’un diabète fragilise la paroi de ces derniers, entraînant une perte d’étanchéité. Il en découle la rupture et l’éclatement de ces vaisseaux. Progressivement, les zones étendues de la rétine ne sont plus oxygénées. Des anomalies structurelles se produisent à la suite d’une diminution métabolique, ce qui entraîne la RD [26].

1.3 Les stades de la rétinopathie diabétique

Afin de déterminer le stade de la RD, des critères sont pris en considération et sont : le type, le nombre et la localisation des lésions présentes (la rétine est divisée en quatre quadrants) [27]. L’évolution de la pathologie peut être classifiée en un certain nombre de stades et qui sont les suivants :

1.3.1 Rétinopathie diabétique non proliférante minime (ou légère)

la rétinopathie non proliférante légère ou autrement dit minime est un stade de la RD caractérisé par la présence de microanévrismes uniquement [27].

La figure ci-dessous représente un cas de rétinopathie diabétique non proliférante minime.

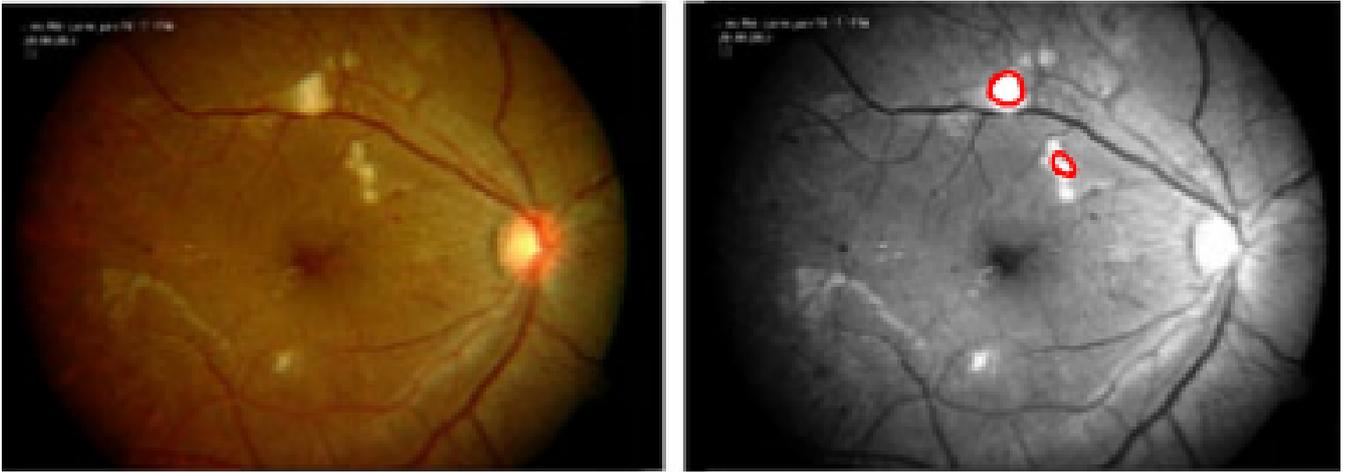


FIGURE 1.2 – Rétinopathie diabétique non proliférante minimale [20]

1.3.2 Rétinopathie diabétique non proliférante modérée

Plus que de simples microanévrismes mais moins qu'une RD non proliférante sévère, ce stade est caractérisé par les symptômes suivants : microanévrismes, exsudats secs, nodules cotonneux et hémorragies rétinienne punctiformes [27].

La figure ci-dessous représente un cas de rétinopathie diabétique non proliférante modérée.

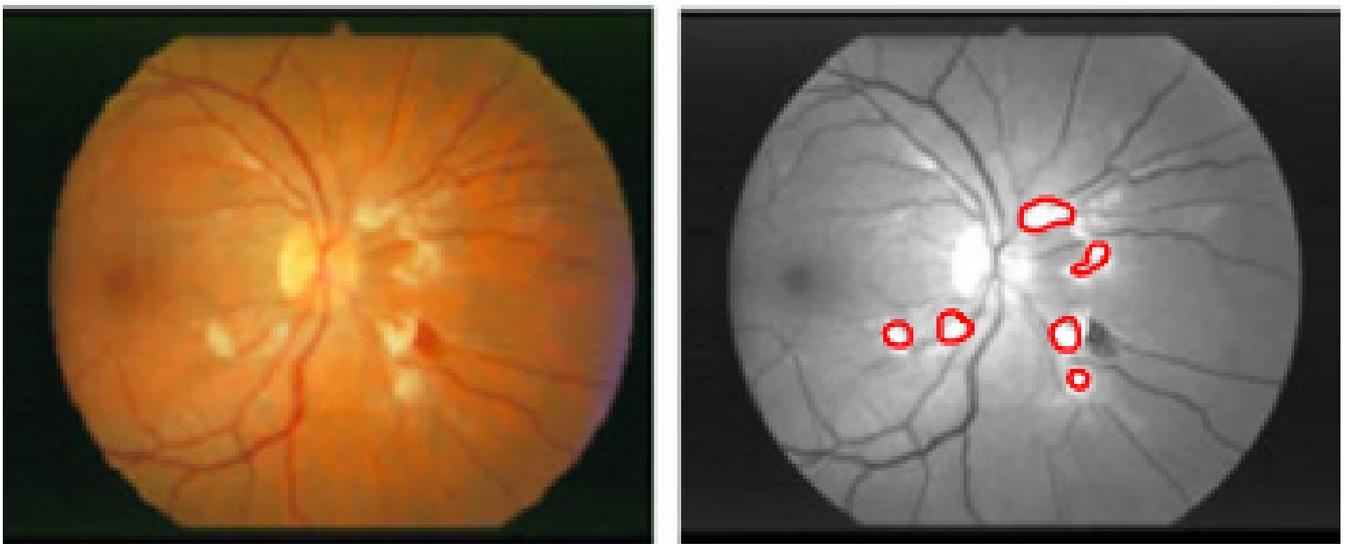


FIGURE 1.3 – Rétinopathie diabétique non proliférante modérée [20]

1.3.3 Rétinopathie diabétique non proliférante sévère

Afin que la RD soit classée comme non proliférante sévère il faut qu'il n'y est pas de signes de RD proliférante et que l'un de ces trois symptômes soit présent [27] :

- Plus de 20 hémorragies intrarétiniennes dans chacun des 4 quadrants.
- Anomalies veineuses certaines dans au moins 2 quadrants.

- AMIR (Anomalies Microvasculaires Intrarétiniennes) proéminent dans au moins 1 quadrant.

La figure ci-dessous représente un cas de rétinopathie diabétique non proliférante sévère.

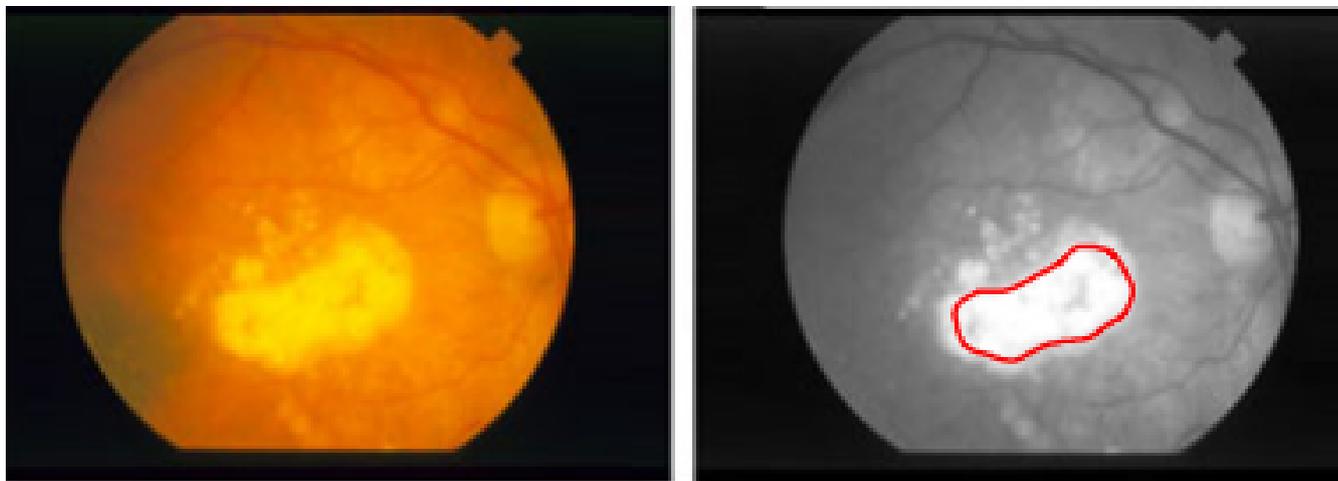


FIGURE 1.4 – Rétinopathie diabétique non proliférante sévère [20]

1.3.4 Rétinopathie diabétique proliférante

Une RD est classée comme proliférante si l'une des conditions suivantes est satisfaite [27] :

- Présence des néo vaisseaux.
- Hémorragies vitrée/ pré rétinienne.

La figure ci-dessous représente un cas de rétinopathie diabétique proliférante.



FIGURE 1.5 – Rétinopathie diabétique proliférante [21]

Conclusion

Dans ce chapitre nous avons présenté et éclairci ce qu'est la rétinopathie diabétique, sa cause, ses stades et les signes cliniques caractérisant chaque stade. L'objectif principal de ce travail est de développer un système multi-agents capable de détecter la présence de cette maladie et cela en analysant des images de fond d'œil. C'est pourquoi, le chapitre suivant portera sur les agents et les systèmes multi-agents.

Chapitre 2

Application des techniques de l'IA dans le diagnostic médical

Introduction

L'intelligence artificielle (IA) est un terme général qui implique l'utilisation d'un ordinateur pour modéliser un comportement intelligent avec une intervention humaine minimale [28]. L'objectif ultime est de créer des programmes informatiques capables de résoudre des problèmes et d'atteindre des objectifs dans le monde aussi bien que les humains[29]. L'IA peut être retrouvée dans plusieurs domaines dont : L'automobile (voiture autonome), la médecine, l'industrie, le marketing, l'éducation et bien plus encore. Dans notre cas nous nous intéressons au domaine de la santé ou autrement dit au domaine médical. Les principaux champs d'application de l'IA en matière de santé sont : la médecine prédictive notamment la prédiction d'une maladie mais aussi l'aide à la décision pour établir un diagnostic médical. Plusieurs techniques de l'IA ont vu le jour, mais dans cette étude nous nous concentrerons sur le Machine Learning et les agents et systèmes multi-agents. Dans ce Chapitre, nous parlerons de l'application des techniques et outils de l'IA, nous nous focaliserons sur les techniques du Machine Learning pour la classification et la prédiction d'une pathologie qu'est la rétinopathie diabétique ainsi que les approches multi-agents pour le diagnostic médical.

2.1 Techniques de l'apprentissage automatique pour la classification et la prédiction de la rétinopathie diabétique

L'ophtalmologie est l'une des spécialités médicales qui fait le plus appel à l'intelligence artificielle pour détecter, diagnostiquer et traiter diverses pathologies oculaires telles que le glaucome, la dégénérescence maculaire et la rétinopathie diabétique. L'apprentissage automatique (machine learning) et l'apprentissage profond (deep learning) représentent les méthodes les plus utilisées dans ce domaines. Dans ce qui suit, nous allons d'abord définir quelques concepts liés au ML avant de présenter quelques travaux récents jugés des plus pertinents qui ont appliqué ces techniques pour la classification et la prédiction de la rétinopathie diabétique.

2.1.1 Machine Learning

Le Machine Learning ou autrement dit l'apprentissage automatique est un sous-domaine de l'IA, né de la volonté d'entraîner des machines à effectuer des tâches habituellement confiées à des êtres humains. Le Machine Learning procure, ainsi, aux systèmes la capacité d'apprendre et d'améliorer automatiquement à partir de l'expérience sans être explicitement programmé.

2.1.2 Types d'algorithmes de Machine Learning

Les algorithmes de Machine Learning peuvent être divisés en 3 types :

2.1.2.1 Apprentissage supervisé

Dans l'apprentissage supervisé, l'algorithme est alimenté avec des données étiquetées. Suite à l'apprentissage, le modèle sera capable de classer de nouvelles données et ce en utilisant une fonction qui relie les entrées à la sortie désirée[30].

2.1.2.2 Apprentissage non supervisé

Dans l'apprentissage non supervisé, l'algorithme est alimenté avec des données non étiquetées. Suite à l'apprentissage, notre modèle trouvera des étiquettes en fonction des similarités que les échantillons partagent. L'algorithme sera capable de classer les nouvelles données en fonction des nouvelles étiquettes créées[30].

2.1.2.3 Apprentissage par renforcement

L'apprentissage par renforcement suit le concept de la méthode des coups et des essais. Le modèle effectue une action sur l'environnement et obtient une réponse en retour de ce dernier. Le système d'apprentissage juge son action comme étant bonne (gratifiante) ou mauvaise (punissable) en fonction de la réponse reçue par l'environnement. Il effectue, en conséquence, un ajustement de ses paramètres. Généralement, l'ajustement des paramètres est poursuivi jusqu'à ce qu'un état d'équilibre se produise[31].

2.1.3 Deep Learning

L'apprentissage profond est un sous-domaine de l'apprentissage automatique qui traite des algorithmes inspirés par la structure et le fonctionnement du cerveau humain. Le Deep Learning est la technique la plus utilisée en apprentissage automatique mais aussi la plus puissante en matière de classification d'où sa présence dans presque toutes les applications et domaines qui utilisent l'apprentissage automatique.

La principale différence entre les algorithmes d'apprentissage profond et d'apprentissage automatique est que les algorithmes d'apprentissage profond se concentrent sur l'apprentissage automatique des caractéristiques, alors que dans l'apprentissage automatique, nous devons extraire les caractéristiques manuellement.

2.1.3.1 Réseaux neuronaux artificiels

Un réseau de neurones artificiels est un système de calcul qui essaye d'imiter ou du moins de s'inspirer des connexions entre les neurones du système nerveux d'un être humain. Les réseaux de neurones artificiels peuvent aussi être appelés systèmes de neurones artificiels. Un réseau neuronal artificiel (multicouche) est constitué de plusieurs couches de neurones qui sont :

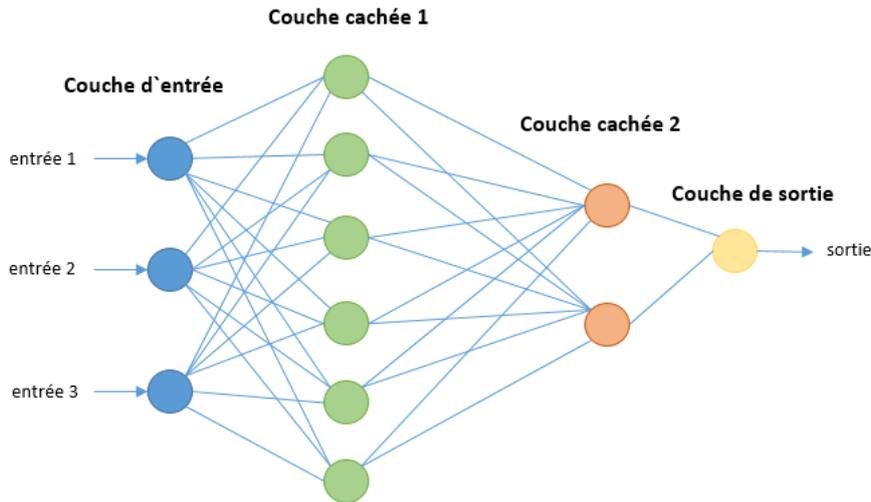


FIGURE 2.1 – Réseaux neuronaux artificiels

2.1.3.1.1 Couche d'entrée : Elle représente la première couche du réseau neuronal, composée de neurones d'entrée. C'est la couche qui reçoit les données.

2.1.3.1.2 Couches cachées : Cette ou ces couches sont situées entre la couche d'entrée et celle de sortie. Dans ces couches, les problèmes sont résolus et ce, car elles permettent d'extraire les caractéristiques à partir des données d'entrées afin de les utiliser par la couche de sortie [32].

2.1.3.1.3 Couche de sortie : Elle représente la dernière couche d'un réseau neuronal artificiel. Elle produit la sortie du programme et donc les résultats calculés par le réseau.

2.1.3.2 Fonctions d'activation

Une notion très importante dans le Deep Learning, Les fonctions d'activation sont des équations mathématiques grâce auxquelles la sortie du réseau de neurones est déterminée. Grâce à ces fonctions, chaque couche du réseau de neurones fonctionne de manière non linéaire. Nombreuses sont les fonctions d'activation et elles peuvent notamment être combinées pour obtenir une meilleure extraction des caractéristiques[33].

Il y a plusieurs fonctions d'activation dont :

2.1.3.2.1 Fonction Hyperbolic tangent : Peut aussi être appelée tanh pour faire court, elle est non linéaire et produit des valeurs entre -1 et 1 [22].

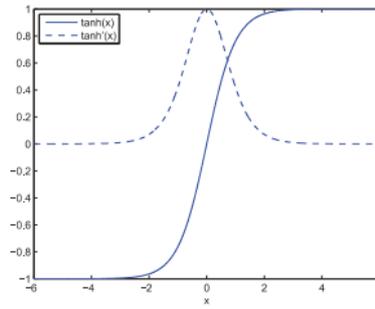


FIGURE 2.2 – Fonction Hyperbolic tangent[22]

2.1.3.2.2 Fonction ReLU : Est l'une des fonctions d'activation les plus connues et utilisées de nos jours. Elle est non linéaire mais aussi, elle a l'avantage de ne pas activer tous les neurones en même temps. Si l'entrée se trouve être une valeur négative alors elle est convertie en 0 et le neurone ne sera donc pas activé. Le fait que seuls quelques neurones sont activés à la fois, fait du réseau un réseau très efficace et rare[22].

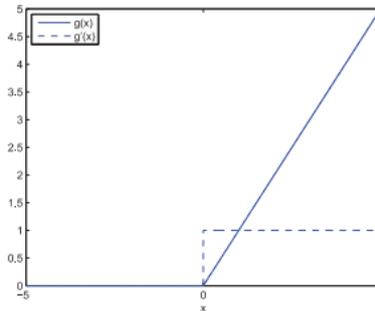


FIGURE 2.3 – Fonction ReLU[22]

2.1.3.2.3 Fonction Sigmoid : Egalement connue sous le nom de fonction logistique, est une fonction d'activation populaire pour les réseaux de neurones. L'entrée de cette fonction est transformée en valeur comprise entre 0 et 1[22].

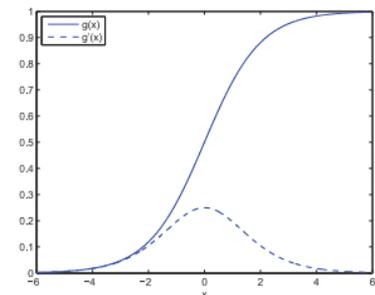


FIGURE 2.4 – Fonction Sigmoid[22]

2.1.3.2.4 Fonction Softmax : Convertit un vecteur de nombres en un vecteur de probabilités, ou les probabilités de chaque valeur sont proportionnelles à l'échelle relative de chaque valeur dans le vecteur. L'utilisation la plus fréquente de cette fonction dans le ML est lorsque

le réseau est configuré pour sortir N valeur, une pour chaque classe dans la classification. Elle est aussi utilisée pour normaliser les sorties, en convertissant des valeurs de somme pondérées en probabilités dont la somme est égale à un. Chacune des valeurs dans la sortie de la fonction softmax est interprétée comme étant la probabilité d'appartenance à chaque classe[34].

2.1.3.3 Réseau neuronal convolutif (CNN)

Un réseau neuronal convolutif (Convolutional deep Neural Network) est un type de réseau profond dans lequel le motif de connexion entre les neurones correspond à des régions qui se chevauchent dans une image. Ce sont les méthodes principales permettant le traitement de données de type "signal", comme les images ou encore le son. Leur fonctionnement est semblable à notre façon de traiter les informations visuelles, grâce à l'empilement de plusieurs perceptrons[33].

2.1.4 Application de l'apprentissage automatique pour la classification de la rétinopathie diabétique

Plusieurs chercheurs se sont intéressés de près à la détection et classification de la rétinopathie diabétique en utilisant des techniques de Machine Learning. La plupart des travaux de recherche ont fait usage des techniques de Deep Learning. Ces travaux peuvent être divisés en plusieurs catégories : Binary classification, multi-level classification, lesion-based classification et Vessel-based classification. Dans notre cas, nous nous sommes intéressés à la classification multi-level qui est une classification multi-classes qui nous permet de classer l'image de fond d'œil en une de ces cinq classes : 0 (Pour NO DR c'est-à-dire pas de présence de rétinopathie diabétique), 1 (Pour mild non-proliférative DR donc rétinopathie diabétique non proliférante légère), 2 (Pour moderate non-proliférative DR donc rétinopathie diabétique non proliférante modérée), 3 (Pour sévère non-proliférative donc rétinopathie diabétique non proliférante sévère), 4 (Pour proliférative DR donc rétinopathie diabétique proliférante).

Dans ce qui suit, nous allons exposer les travaux, que nous avons jugés les plus intéressants, et qui ont traité la problématique de la détection et classification multi-level de la rétinopathie diabétique.

2.1.4.1 Réseaux neuronaux convolutifs pour la rétinopathie diabétique[1]

H.Pratt et al.[1] ont proposé une méthode basée CNN pour la classification des images du dataset kaggle [35] en 5 niveaux de rétinopathie diabétique. Afin de classer les images, ils ont d'abord entamé une phase de prétraitement où ils ont appliqué une normalisation des couleurs et un redimensionnement en 512×512 pixels. Le CNN proposé contient : 10 couches CONV, 8 couches MaxPooling, 3 couches FC. La fonction softmax a été utilisée comme classificateur pour 80 000 images testes. Ce CNN a d'abord été pré-entraîné sur 10 290 images jusqu'à ce qu'il atteigne un niveau significatif et ce pour obtenir un résultat de classement rapide. Après 120 époques d'entraînement sur les images initiales, il a été formé sur 78000 images pendant 20 autres époques. Pour réduire l'overfitting, ils ont utilisé les méthodes de L2 regularisation et Dropout. Avec cette solution, ils ont obtenu une spécificité de 95 %, une accuracy de 75 % et une sensibilité de 30 %. Cependant, le CNN ne détecte pas les lésions dans les images et un

seul dataset a été utilisé pour l'évaluation.

2.1.4.2 Classification d'images de rétinopathie diabétique à l'aide de modèles d'apprentissage profond[2]

S.Dutta et al.[2] détectent et classent les images du dataset kaggle [35] en 5 niveaux de rétinopathie diabétique. Ils se sont concentrés sur l'obtention d'un modèle optimal par l'apprentissage automatique, qui comprend le Back propagation neural network (BNN), le Deep Neural Network (DNN) et le Convolutional Neural Network (CNN). 2000 images du dataset Kaggle [35] ont été utilisées. Elles ont d'abord été redimensionnées en 300×300 pixels puis converties en niveaux de gris. Avant d'envoyer les images aux réseaux de neurones, des filtres ont d'abord été appliqués dont : edge detection, median filter, morphological processing, binary conversion. Le BNN utilisé comprend plusieurs couches d'unités de calcul reliées entre elles par une voie de feed forward, une fonction d'activation "sigmoïde" et l'algorithme traditionnel "Back Propagation", il a été entraîné pendant 268 époques. La méthode de DNN a trois couches entièrement connectées (FC), deux couches d'activation avec "ReLU" comme fonction d'activation, pour construire le modèle, les fonctions d'activation "Tanh" et "Sigmoid" sont également utilisées, mais "ReLU" est plus performant que les deux autres approches. Tandis que le CNN était le modèle VGGnet [36], qui est structuré de couches CONV qui effectuent une convolution 3×3 avec stride :1 et pad :1, et de couches POOL qui effectuent un maxpooling 2×2 avec stride :2, la fonction d'activation utilisée est "ReLU" et les couches convolutionnelles sont suivies par Maxpool utilisé. Les résultats montrent que le DNN est meilleur que le CNN et le BNN. Cela dit, quelques images seulement ont été utilisées pour l'entraînement, ces réseaux n'ont pas pu apprendre plus de caractéristiques et une seule base de données a été utilisée pour l'évaluation.

2.1.4.3 Classification des stades de la rétinopathie diabétique à l'aide de réseaux de neurones convolutifs [3]

X. Wang et al.[3] ont étudié les performances de trois architectures pré-entraînées valables de CNN qui sont : VGG16 [36], AlexNet[37] et InceptionNet V3[38] pour détecter les 5 stades de rétinopathie diabétique. 166 images représentatives et de haute qualité ont été choisies dans le jeu de données Kaggle [35] et ont constitué le jeu de données qu'ils ont utilisé pour entraîner les modèles de classification qu'ils présentent. Ils ont d'abord effectué un prétraitement en redimensionnant les images en 224×224 pixels pour le VGG16 [36], 227×227 pixels pour le AlexNet[37] et 299×299 pixels pour InceptionNet V3[38]. Les images en ensuite été envoyées au modèles de classification pour qu'ils soient entraînés et ce en utilisant un processus de validation croisée. Ils ont obtenu des précisions moyennes (accuracy) de 50.03% pour VGG16[36], 37.43% pour AlexNet [37] et 63.23% pour InceptionNet V3[38]. Cependant, ils ont entraîné les réseaux avec un nombre d'images limité, ce qui empêche ces CNN d'apprendre plus de caractéristiques, les images ont besoin de plus de fonctions de prétraitement pour les améliorer et un seul dataset a été utilisé pour évaluer leur étude.

2.1.4.4 Filtrage automatique des images du fond d'œil à l'aide d'une combinaison de réseaux neuronaux convolutionnels et du hand-crafted features[4]

B. Harangi et al.[4] ont proposé une combinaison de AlexNet [37] qui est un CNN pré-entraîné et du hand-crafted features pour classer les 5 stades de la rétinopathie diabétique. Pour construire leur réseau et ainsi fusionner le CNN et Hand-Crafted Features, ils ont étendu la couche FCfuse en lui ajoutant un vecteur de 68 dimensions contenant les caractéristiques traditionnelles pour obtenir 4164 caractéristiques. Ensuite, la couche FCclasse de 4164×5 (respectivement 4164×3) est considérée pour la tâche de classification RD (respectivement DME). De cette façon, la pondération finale FCclass des caractéristiques CNN/traditionnelles et les 4096 caractéristiques AlexNet[37] peuvent être entraînées par rétropropagation. Le modèle proposé a été entraîné sur le jeu de données Kaggle [35] et testé sur le jeu de données IDRiD [39]. Et ils ont obtenu une accuracy de 90.07%.

2.1.4.5 Évaluation diagnostique des algorithmes d'apprentissage profond pour le dépistage de la rétinopathie diabétique[5]

T. Li et al.[5] ont détecté les stades de la rétinopathie diabétique dans leur base de données [40] en affinant les réseaux de neurones pré-entraînés GoogleNet[41], DenseNet-121[42], VGG16 [36], ResNet-18[43], SE-BN-Inception[44] disponibles. Ils avaient pour objectif d'évaluer ces méthodes en situation clinique et c'est pour cela qu'ils ont recueilli 13 673 images de fond d'œil de 9598 patients et ainsi ils ont construit leur propre jeu de données nommé DDR [40]. Pendant le prétraitement, les images ont été recadrées et redimensionnées en 224×224 pixels, augmentées et échantillonnées pour équilibrer les classes. Les résultats expérimentaux montrent que le réseau SE-BN-Inception[44] présente la meilleure accuracy pour la classification de la rétinopathie diabétique et ce avec 0.8284. D'après les auteurs, DDR [40] est le premier jeu de données d'images de fond d'œil représentatif d'une population chinoise mais aussi le deuxième plus grand pour la classification de la rétinopathie diabétique.

2.1.4.6 Architecture Alexnet modifiée pour la classification des images de rétinopathie diabétique[6]

T. Shanthi et al.[6] ont détecté les stades de la rétinopathie diabétique en utilisant une architecture pré-entraînée AlexNet[37]. La détection a été effectuée sur le jeu de données Messidor[45] qui comporte environ 1190 images de fond d'œil en couleur avec des annotations dans un fichier Excel, aussi il a été réparti en quatre sous-ensembles différents : rétine saine, stade DR 1, stade DR 2 et stade DR 3. Le jeu de données a été divisé en deux, 70% des images de fond d'œil ont été utilisées pour l'entraînement de l'architecture AlexNet[37] et les autres 30% des images de fond d'œil ont été utilisées pour évaluer la performance de l'algorithme. Les images ont été redimensionnées en 259×259 pixels et le canal vert a été extrait avant de les envoyer au CNN. Ils ont obtenu à l'aide de l'architecture proposée, une précision de classification des images du jeu de données Messidor[45] : 96.6% pour la rétine saine, 96.2% pour la rétinopathie diabétique de stade 1, 95.6% pour la rétinopathie diabétique de stade 2 et 96.6% pour la rétinopathie diabétique de stade 3.

2.1.5 Application de l'apprentissage automatique pour la prédiction de la rétinopathie diabétique

Prédire la rétinopathie diabétique peut représenter un atout majeur dans la médecine ophtalmologique. En effet, prédire le risque d'apparition de la rétinopathie diabétique peut aider dans la prévention de cette dernière et ce en prenant les mesures nécessaires pour l'éviter. Plusieurs travaux liés à la prédiction du risque de rétinopathie diabétique ont été réalisés, nous allons citer certains d'entre eux dans ce qui suit :

2.1.5.1 Calculateurs de risque de rétinopathie dans la prédiction de la rétinopathie diabétique menaçant la vue dans le diabète de type 2 : Une sous-étude FIELD[7]

BN Rao et al.[7] ont eu pour objectif l'évaluation de l'algorithme de risque d'Aspelund et al. [46] pour prédire la rétinopathie diabétique menaçant la vue (STDR) dans le diabète de type 2 (T2D), et développer un nouveau modèle de prédiction de la STDR. Ils ont utilisé l'algorithme d'Aspelund et al. [46] pour calculer le risque de STDR à 2 et 5 ans à partir de variables de base chez 1012 participants à la sous-étude ophtalmologique "Fenofibrate Intervention and Event Lowering in Diabetes (FIELD)" tels que diabète (1 ou 2), le sexe, la durée du diabète au départ, l'HbA1c concomitante, la pression artérielle systolique et si non STDR était présent ; comparé au statut STDR à l'essai, et une analyse des caractéristiques d'exploitation du récepteur a été effectuée. À l'aide d'une régression logistique multivariable, les facteurs de risque traditionnels et l'allocation de fénofibrate en tant que prédicteurs de la STDR ont été évalués, et des estimations ajustées de l'optimisme basées sur le bootstrap ont été calculées pour la performance prédictive. La STDR s'est développée chez 28 participants et l'algorithme d'Aspelund et al. [46] a prédit avec précision la STDR à 2 et 5 ans avec une aire sous la courbe (AUC) de 0,86 et 0,86 , respectivement. Dans le second modèle, les facteurs de risque STDR étaient la présence d'une RD au départ, l'HbA1c et le sexe masculin, tandis que le fénofibrate était protecteur. Ce modèle présentait une excellente capacité de discrimination (AUC = 0,89).

2.1.5.2 Évaluation individualisée du risque de rétinopathie diabétique et optimisation des intervalles de dépistage : une approche scientifique pour réduire les coûts des soins de santé[8]

SH. Lund et al.[8] ont eu pour objectif de valider un algorithme mathématique qui calcule le risque de progression de la rétinopathie diabétique dans une population diabétique de 9690 personnes en Angleterre qui participent au programme national anglais de dépistage du diabète. L'algorithme [46] utilisé a été conçu à l'origine pour estimer le risque de développer une rétinopathie menaçant la vue (STR, soit un œdème maculaire diabétique (OMD) ou une rétinopathie diabétique profilée (RDP)), mais, il a été adapté afin de prédire le risque d'apparition d'une rétinopathie de stade R2, R3A ou M1, au lieu de STR et ce en calculant un coefficient de calibration pour chaque stade de rétinopathie, ainsi que pour le sexe et le type de diabète. Ce coefficient correspondait à la proportion observée de chaque résultat divisée par l'estimation du risque moyen pour STR. Une estimation du risque de développer R2, R3A ou M1 a été calculée en multipliant l'estimation du risque de l'algorithme par le coefficient de calibration

correspondant. L'algorithme prédit de manière fiable le risque de progression de la rétinopathie diabétique vers R2, R3A et M1 et les profils de risque de la cohorte montrent qu'une grande majorité présente un risque inférieur à 3,4 % de progression entre les stades en 1 an et un petit groupe à haut risque est identifié.

2.1.5.3 Modèle prédictif et analyse du risque de rétinopathie diabétique à l'aide de l'apprentissage automatique : une étude de cohorte rétrospective en Chine[9]

Li W, et al.[9] avait pour objectif d'étudier les facteurs de risque de rétinopathie diabétique (RD) et les modèles prédictifs par apprentissage automatique à l'aide d'un vaste ensemble de données. Ils ont proposé un modèle de prédiction construit sur la base de l'algorithme XGBoost. Les informations sur 32 452 patients hospitalisés atteints de diabète sucré de type 2 (DT2) ont été extraites du système de dossiers médicaux électroniques du 1er janvier 2013 au 31 décembre 2017. Afin d'améliorer l'utilisation des données, les données manquantes ont dû être interpolées avec la méthode de l'interpolateur à k voisins les plus proches (KNNI). Soixante variables ont été retenues pour l'analyse de base et 17 variables optimales ont été sélectionnées par élimination récursive des caractéristiques. La RD est survenue chez 2038 (6,28%) patients atteints de DT2. Ils ont comparé les performances de XGBoost avec celles de trois autres techniques d'apprentissage automatique populaires : Logistic regression (LR), random forest (RF) et Support vector machine (SVM) et le modèle XGBoost a été identifié comme le meilleur modèle de prédiction avec la plus grande AUC (aire sous la courbe, 0,90).

2.1.5.4 Prévision de la rétinopathie diabétique et identification de caractéristiques biomédicales interprétables à l'aide d'algorithmes d'apprentissage automatique [10]

H. Tsao et al.[10] ont eu pour objectif de construire un modèle de prédiction de la RD chez les diabétiques de type 2 en utilisant des techniques d'exploration de données. Le but était d'obtenir une performance de prédiction précise, mais aussi de générer un modèle interprétable pour la pratique clinique. Pour prédire la rétinopathie diabétique, ils ont d'abord extrait les variables démographiques, les résultats des tests de laboratoire, les antécédents familiaux de diabète et les habitudes d'exercice des patients. Ensuite, ils ont appliqué différents algorithmes d'apprentissage automatique (Les arbres de décision, les machines à vecteurs de support, la régression logistique et les réseaux de neurones artificiels) pour obtenir une prédiction précise et identifier de nouveaux facteurs de risque. Le logiciel "SAS Enterprise Miner version 12.1" a été utilisé pour générer des modèles de prédiction. Les résultats expérimentaux ont montré que les performances de prédiction des machines à vecteurs de support étaient meilleures que celles des autres algorithmes d'apprentissage automatique et qu'elles atteignaient 79,5 % en termes de précision et 0,839 d'aire sous la courbe. Leur méthode identifie l'utilisation de l'insuline et la durée du diabète comme de nouvelles caractéristiques interprétables pour prédire la rétinopathie diabétique.

2.1.5.5 Prédiction du risque de rétinopathie diabétique à partir de l'examen du fond d'œil en utilisant l'apprentissage clairsemé : une étude transversale[11]

Oh et al.[11] ont développé et validé des modèles d'apprentissage clairsemés ou autrement dit Sparse Learning dans le but d'identifier le risque de RD chez les patients diabétiques. Ces modèles d'apprentissage clairsemés développés ont été le Ridge, elastic net et LASSO, ce sont des outils largement utilisés en bio-informatique [47]. Afin de développer leurs modèles de prédiction du risque de rétinopathie diabétique, les données de 490 personnes ayant participé à l'enquête KNHANES V-1 ont été utilisées. Cet ensemble de données a été divisé aléatoirement en ensemble d'entraînement (327 patients) et en ensemble de validation interne (164 patients). La performance du modèle de prédiction a été évaluée sur les données indépendantes de 562 participants sélectionnés de l'enquête KNHANES V-2, considérées comme étant un ensemble de données de validation externe. Les résultats obtenus ont montré que LASSO a prédit plus efficacement la RD. En effet, lors de la validation interne, il a montré une AUC de 0.81 et une précision de 73.6% et lors de la validation externe il a obtenu une AUC de 0.82 et une précision de 75.2%. Ce qui montre qu'il est significativement supérieur aux indicateurs traditionnels.

2.2 Approches multi-agents pour le diagnostic médical

Les travaux de recherche récents portent un intérêt considérable aux systèmes multi-agents (SMA). Les SMA représentent un outil majeur dans la résolution de problèmes complexes et cela en les décomposant en multiples petites tâches. Dans cette section, nous exposerons des définitions et des généralités sur les agents et systèmes multi-agents, leurs avantages ainsi que leur application dans le domaine du Healthcare.

2.2.1 Définitions

Dans un système multi-agents, l'entité principale est appelée « Agent ». Le terme « Agent » peut être retrouvé dans plusieurs domaines mais ne reconnaît pas une définition acceptée unanimement. Néanmoins, dans ce travail nous prendrons en considération cette définition : [48]

Définition 2.2.1. Un agent est une entité autonome qui évolue dans un environnement et est capable d'agir à l'aide d'actionneurs. Cette entité utilise des capteurs pour une perception partielle et communique avec d'autres agents de cet environnement. Son comportement vise à satisfaire ses besoins et ses objectifs à partir de ses observations, de ses connaissances, son historique d'actions et ses interactions avec les autres agents.

La définition ci-dessus inclut trois mots-clés capitales :

- **Entité** : Cela désigne le type de l'agent. Un agent peut être une entité physique : un robot, une voiture, un avion, ou une entité virtuelle : un composant logiciel, un module informatique.
- **Environnement** : L'environnement désigne le lieu où se trouve l'agent, et se caractérise par tout ce qui n'est pas lui. Par exemple, ce qui peut former l'environnement d'un agent

taxi sont : les routes, les autres automobiles, les piétons, les clients, etc... Les informations captées sur l'environnement sont utilisées par l'agent pour prendre des décisions.

- **Action** : Afin de modifier l'environnement, un agent peut effectuer des actions. Par exemple, un agent taxi accélère alors la position du taxi change.

Un agent peut être caractérisé par ses actions ou autrement dit son comportement qui est analysable sans connaître les détails d'implémentations. Les propriétés suivantes caractérisent les agents et leurs permettent de résoudre des tâches complexes :

- **Réactivité** : Un agent peut percevoir l'environnement où il est situé et ainsi répondre en temps réel aux changements, cela veut dire qu'un agent est capable d'agir, de réagir et de modifier l'environnement.
- **Proactivité** : Un agent est dirigé par ses buts, ce qui fait qu'il est doté d'un comportement orienté but lui permettant de prendre l'initiative. Cette capacité implique qu'un même agent peut prendre des actions différentes lorsqu'il est placé dans des environnements différents.
- **Sociabilité** : Un agent n'est pas isolé mais fait plutôt partie d'une société d'agents. Il est capable d'interagir avec d'autres agents ou utilisateurs pour partager leurs connaissances et demander des informations afin d'améliorer leurs performances dans la réalisation de leurs objectifs.
- **Autonomie** : Un agent est autonome et cela veut dire qu'il est capable d'agir sans l'intervention directe d'un autre agent ou d'un utilisateur. Il a la capacité de contrôler ses propres actions ainsi que son état interne, et de ce fait, un agent peut refuser des requêtes de ces derniers.

Définition 2.2.2. Un système multi-agents est un système composé d'un ensemble d'agents autonomes qui interagissent, communiquent et coopèrent pour résoudre collectivement un problème ou élaborer une expertise sans l'intervention d'un externe au système afin d'atteindre leurs objectifs [48].

2.2.2 Avantages des systèmes multi-agents

Les systèmes multi-agents offrent beaucoup d'avantages dans le développement des systèmes complexes, étant donné que le paradigme agent est considéré comme le paradigme idéal pour le développement de ce genre de systèmes. Nous pouvons citer de nombreux avantages dont : [49]

- **La modularité** : Un système multi-agents est composé d'un ensemble d'entités (les agents) qui sont autonomes et indépendantes l'une des autres. L'autonomie des agents signifie que chaque agent peut contrôler son état et son comportement. L'interaction entre les agents est essentiellement basée sur l'échange de messages. Par conséquent, le couplage entre les agents est faible. En plus, plusieurs concepts de base des systèmes multi-agents contribuent à la modularité de ces derniers tel que le concept « organisation ». En effet, les organisations multi-agents sont modélisées, généralement, par des sous-organisations (des groupes, des coalitions, des hiérarchies...), ce qui rend les agents plus modulaires.

- **La réutilisabilité** : La réutilisabilité est une conséquence directe de la modularité des systèmes multi-agents. Les systèmes multi-agents peuvent atteindre la réutilisabilité à plusieurs niveaux. Naturellement, l'agent est conçu pour être une brique aussi réutilisable que possible. Ainsi, on peut réutiliser les composants d'un agent pour développer un autre agent. Le mécanisme de raisonnement d'un agent est généralement une composante indépendante du domaine d'application ce qui fait qu'il est possible, de développer un seul mécanisme et de le réutiliser ensuite pour le développement des agents.
- **La facilité de maintenance** : Le développement modulaire des systèmes multi-agents simplifie la maintenance de ces systèmes. En effet, ils sont conçus comme des entités indépendantes avec un couplage faible, ce qui fait que la modification d'une entité a un impact faible sur les autres entités et c'est ce qui rend la maintenance facile.
- **La fiabilité** : On désigne par la fiabilité, la capacité d'un système à continuer son fonctionnement en dépit de l'apparition des pannes ou des erreurs dans l'une ou plusieurs de ces parties. Les systèmes multi-agents peuvent contribuer à la fiabilité des logiciels grâce à l'autonomie des agents, en effet, un agent est une entité autonome dans le sens où elle peut atteindre ses objectifs sans l'intervention des autres entités. De plus, un agent peut changer son comportement dans les situations imprévisibles et cela grâce à sa flexibilité.
- **L'efficacité** : L'efficacité des logiciels est mesurée en fonction de ressources consommées afin de résoudre un problème. Les systèmes multi-agents représentent un modèle d'exécution distribué dans lequel les ressources sont distribuées. Ce partage permet une utilisation rationnelle des ressources. En plus certains types d'agents permettent une meilleure exploitation de certaines.
- **L'adaptation à la réalité** : Le paradigme agent permet une modélisation fidèle de phénomènes réels, étant donné que les caractéristiques des agents, comme l'autonomie, la flexibilité et la sociabilité, représentent des caractéristiques intrinsèques de plusieurs systèmes réels (comme les sociétés humaines, les systèmes biologiques, les colonies d'insectes...)
- **Les modes d'interaction sophistiqués** : Les systèmes multi-agents supportent des modes d'interaction sophistiqués comme la négociation, la coopération et la collaboration.

2.2.3 Application des systèmes multi-agents dans le domaine du Healthcare

Les systèmes multi-agents (SMA) peuvent être considérés comme une technologie appropriée pour le développement d'applications de santé. En effet, l'utilisation de composants hétérogènes et faiblement couplés, la gestion dynamique et distribuée des données et la collaboration à distance entre les différentes entités sont souvent considérés comme les exigences les plus pertinentes. Dans ce qui suit, nous allons présenter les travaux récents les plus notables qui proposent l'utilisation des systèmes multi-agents dans le healthcare.

2.2.3.1 Agent d'apprentissage profond amélioré et robuste pour la détection préliminaire de la rétinopathie diabétique[12]

G.Saxena et al. [12] ont proposé un agent d'apprentissage profond amélioré et robuste pour la détection préliminaire de la rétinopathie diabétique.

Le problème a été défini comme étant un problème de classification binaire qui a été abordé en développant un agent intelligent capable de trouver des modèles dans les images de fond d'œil et classer les images. Une image est classée « positive » (présence d'une rétinopathie diabétique) si elle appartient aux grades 1, 2, 3 ou 4 (Mild, Moderate, Severe et Proliferate respectivement) et en « négative » (absence de rétinopathie diabétique) si elle appartient au grade 0 (No-DR). Afin de développer l'agent, Plusieurs modèles d'apprentissage automatique (CNN) ont été construits qui ont été entraînés et validés sur l'ensemble de données publiques Eye-PACS. Une fois entraînés, ces modèles sont testés sur un autre ensemble de données publiques Messidor 1 et 2. Les résultats obtenus avec cette approche ont été : une sensibilité de 84.74%, une spécificité de 89.65% et une AUC de 0.927 pour la dataset EyePACS et une sensibilité de 88.84%, une spécificité de 89.92% et une AUC de 0.958 pour Messidor-1, tandis que pour Messidor-2 les résultats étaient les suivants : une sensibilité de 81.02%, une spécificité de 86.09% et une AUC de 0.92. L'un des avantages de cette approche est le traitement par lots des images ainsi que la réduction du temps d'inférence du modèle (1,5 s), ce qui rend le processus de dépistage efficace et répond à l'exigence d'un dépistage.

2.2.3.2 Détection d'Alzheimer basée sur des systèmes multi-agents : Un environnement intelligent de traitement d'images[13]

H.Allioui et al.[13] proposent un système de segmentation d'images IRM pour la détection de la maladie d'Alzheimer composé d'un ensemble d'agents autonomes et qui a pour objectif d'améliorer les performances de segmentation afin de faciliter la détermination des lésions ou tumeurs cérébrales. Dans cette méthode, l'image passe d'abord par une phase de prétraitement. Après élimination du bruit et super-pixellisation de l'image, une carte primaire des régions pour la distribution des points de départ est obtenue où les agents sont positionnés et chaque agent recherche les pixels similaires à son pixel de départ. Les agents de contour et les agents de région coopèrent pour obtenir des résultats de segmentation fiables. En cas de situation conflictuelle, les agents négocient pour résoudre le problème. Après ces opérations, les données de segmentation sont extraites, évaluées et validées afin d'obtenir une image segmentée correcte, ainsi que toutes les statistiques nécessaires. Les données testées ont été téléchargées à partir de la base de données NACC (National Alzheimer's Coordinating Center). L'approche proposée offre des résultats optimaux avec très peu d'efforts de calcul. Ceci explique l'efficacité du système proposé dans la reconnaissance et la segmentation des images de la maladie d'Alzheimer. Un autre avantage de cette méthode est que la maladie d'Alzheimer peut être identifiée à un stade précoce ou initial.

2.2.3.3 Contrôle optimisé pour la segmentation d'images médicales : accords améliorés de systèmes multi-agents à l'aide de l'optimisation par essaims (Particle Swarm Optimization)[14]

H. Alliou et al.[14] Proposent une méthode qui consiste à combiner un protocole de contrôle amélioré MAS et l'algorithme d'optimisation par essaims particuliers PSO pour la segmentation d'images médicales 2D et 3D. Cette méthode utilise des techniques de segmentation basées sur la coopération des systèmes multi-agents. Le MAS implémente l'algorithme PSO pour garantir un accord mutuel. Le processus de segmentation comprend sept étapes qui sont : préparation, initialisation, évaluation PSO, mise à jour PSO, évaluation MAS, interactions des agents MAS et mise à jour MAS. Les expériences sont menées en utilisant le jeu de données de démonstration d'imagerie médicale SOFTENTA (2019). Le jeu de données illustré dans cet article contient 501 images IRM à segmenter par le MAS proposé. Les tests ont montré que les résultats obtenus avec le MAS-PSO présenté dans cet article sont meilleurs que ceux obtenus avec les autres approches. En effet, MAS-PSO réduit le temps de réalisation de l'accord mutuel ce qui améliore les effets du contrôle MAS, rend aussi les courbes plus lisses, donne le temps d'attente le plus court et il a pu réduire ce temps de 40% en moyenne, il offre aussi de meilleures performances avec un grand nombre d'agents.

2.2.3.4 Un modèle d'optimisation multi-agents piloté par les bords pour la détection des maladies infectieuses[15]

Y. Djenouri et al. [15] présentent un nouveau cadre intelligent pour la détection des maladies infectieuses en explorant divers paradigmes émergents et intelligents où ils proposent de nouvelles architectures d'apprentissage profond, telles que les réseaux d'intégration d'entités, la mémoire à long et court terme et les réseaux neuronaux à convolution, pour l'apprentissage précis de données médicales hétérogènes dans l'identification des maladies infectieuses. Le système multi-agents est également consolidé pour augmenter les comportements d'autonomie du cadre proposé, où chaque agent peut facilement partager les résultats d'apprentissage dérivés avec les autres agents du système. Une expérimentation intensive a été établie sur trois bases de données médicales avec différentes représentations de données : données brutes, séries temporelles et images qui sont : Respiratory Sound Data, Real-time COVID-19 Data et Corn Leaf Infection Data. Le taux de détection atteint 98% pour le traitement de cas d'utilisation réels.

2.2.3.5 Une approche multi-agents d'apprentissage par renforcement profond pour améliorer la segmentation des images tomodensitométriques COVID-19[16]

H.Alliou et al.[16] présentent une nouvelle méthode d'extraction de masques basée sur l'apprentissage par renforcement profond multi-agents afin de minimiser l'extraction manuelle à long terme des masques et d'améliorer les cadres de segmentation des images médicales. Cette méthode utilise une version modifiée du Deep Q-Network pour permettre au détecteur de masques de sélectionner des masques à partir de l'image étudiée. La méthode a été utilisée sur des images de tomographie informatisée (CT) afin d'extraire les caractéristiques visuelles des zones infectées par COVID-19 et fournir un diagnostic clinique précis tout en optimisant le test de diagnostic pathologique et ainsi gagner du temps. Aucune des principales archives ne

disposait d'une collection substantielle de données étiquetées COVID-19, ce qui les a obligés à s'appuyer sur diverses sources d'images de cas normaux, de pneumonie et de COVID-19. Les résultats obtenus ont été les suivants : une précision de 97,12 % avec un Dice de 80,81 %, une sensibilité de 79,97 %, une spécificité de 99,48 %, une précision de 85,21 %, un score F1 de 83,01 %, une métrique structurelle de 84,38 % et une erreur absolue moyenne de 0,86 %.

2.2.3.6 Utilisation d'un système multi-agents pour la détection des microanévrismes dans les images de fond de l'œil[17]

C. Pereira et al.[17] ont proposé une nouvelle approche, basée sur un modèle multi-agents, pour la segmentation des microanévrismes dans les images de fond d'œil. Une phase de pré-traitement des images a été d'abord entreprise où l'image du plan vert a été choisie, un grand filtre médian a été appliqué sur cette image afin d'obtenir une image d'arrière-plan estimée. Cette image d'arrière-plan estimée a ensuite été soustraite de l'image verte d'origine. Un filtre gaussien a été appliqué pour mettre en valeur les petites structures sombres. Ils ont ensuite appliqué un filtre de Kirsch modifié [50] pour obtenir des bords d'une épaisseur de deux pixels. Le système proposé est composé d'un ensemble d'agents réactifs : les agents d'exploration (EA) et les agents de région (RA). Ces agents sont dynamiquement lancés et détruits sans contrôle externe. Ils ont la capacité d'extraire de l'environnement les informations dont ils ont besoin et d'agir de manière autonome sur celles-ci. L'environnement englobe les informations préparées (images d'entrée), leurs caractéristiques, et les résultats. La segmentation et l'exploration de l'image sont commandées par les comportements de l'agent, l'EA explore l'environnement et lance un RA lorsqu'il trouve une région d'intérêt. Le RA segmente et analyse ensuite sa région. En outre, les AR négocient entre elles pour tenter de fusionner des régions. Les résultats obtenus étaient compétitifs et le score final de l'approche proposée était de 0,240 dans le défi en ligne de la rétinopathie.

2.2.3.7 Une approche multi-agents pour la prédiction personnalisée de l'hypertension artérielle[18]

S.Abrar et al.[18] ont proposé un système multi-agents intégré pour la prédiction personnalisée du risque d'hypertension. Ce système détecte et calcule les valeurs manquantes dans les séries chronologiques et fournit une prédiction personnalisée du risque d'hypertension. Étant donné que le système proposé a été développé suivant une approche multi-agents, les tâches ont donc été divisées et distribuées sur quatre agents qui sont : user agent, the data processing agent, the blood pressure prediction agent et the blood pressure status agent. Cette solution peut prendre périodiquement la pression sanguine de l'utilisateur et ainsi réaliser la prédiction pour les prochaines 24 heures mais aussi estime le risque d'hypertension sur 4 ans afin d'éviter l'apparition de l'hypertension, le score de risque d'hypertension est calculé en utilisant le Framingham hypertension risk estimator. Le modèle proposé a été évalué sur des données de pression sanguine recueillies auprès de la population malaisienne et utilisant l'erreur absolue moyenne, l'erreur quadratique moyenne et La racine de l'erreur quadratique moyen.

Conclusion

Dans ce chapitre, nous avons présenté des techniques de l'IA dont l'utilisation ne cesse d'augmenter. Ces techniques sont Le Machine Learning et les systèmes multi-agents. Nous avons exposé des généralités et des définitions ainsi que des états de l'art relatifs à l'application de ces deux outils de l'IA dans le domaine médical. Le chapitre suivant sera dédié à notre proposition.

Chapitre 3

Un modèle multi-agents basé sur le Deep Learning pour la prédiction de la rétinopathie diabétique

Introduction

Plusieurs travaux scientifiques ont couplé la recherche sur la technologie des agents et les techniques de l'apprentissage automatique. Une telle combinaison est motivée par les difficultés rencontrées par les deux communautés et par la nécessité de développer des approches de traitement des données plus avancées et des systèmes plus intelligents. En effet, Les techniques de l'apprentissage automatique ont fait leur preuves dans diverses domaines, cependant elles s'avèrent peu appropriées quand il s'agit d'un environnement complexe et dynamique qui peut affecter la performance globale du système. Les systèmes multi-agents, quant à eux, ont révélé leur potentiel d'améliorer les systèmes distribués, parallèles et concurrents de diverses manières. Un système d'apprentissage automatique basé sur des agents contribue positivement au processus d'apprentissage automatique notamment dans les cas où les sources de données sont distribuées ou distantes. Les agents doivent alors collaborer pour intégrer des informations provenant de diverses sources locales.

Dans ce travail, nous proposons de combiner ces deux approches afin de concevoir un modèle capable de collecter les données et d'user des techniques de l'apprentissage automatique pour les traiter et les analyser afin de prédire l'apparition de la rétinopathie diabétique chez un sujet diabétique.

Dans ce chapitre nous présenterons notre propre proposition. Nous décrirons de façon précise notre modèle multi-agents et nous détaillerons les différentes interactions entre les agents qui le composent.

3.1 Architecture du modèle multi-agents proposé

Le but de notre recherche est de développer un système multi-agents capable de prédire la rétinopathie diabétique pour éviter l'apparition de cette dernière.

Notre système a été développé suivant une approche multi-agents de tel sorte à ce que les tâches soient divisées et distribuées sur trois agents :Agent utilisateur, agent classification et agent prédiction.

La figure 3.1 illustre l'architecture de notre modèle système multi-agents.

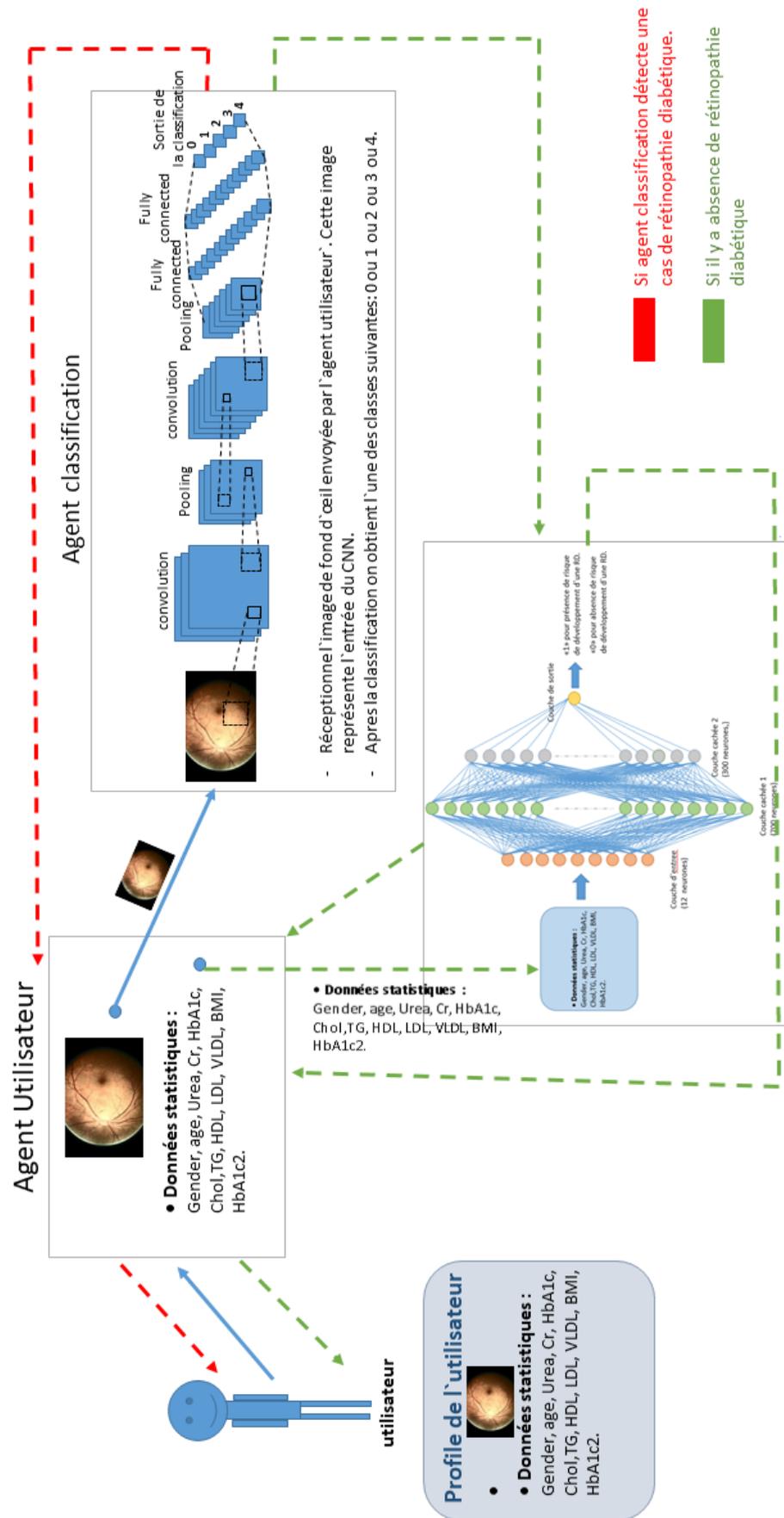


FIGURE 3.1 – Architecture du modèle multi-agents proposé

Dans ce qui suit une description approfondie de chaque agent du système multi-agents

3.1.1 Rôles et architecture interne des agents

Nous allons décrire, ci-dessous, le rôle de chaque agent mais aussi leurs architectures internes.

3.1.1.1 Agent utilisateur

- **Rôle de l'agent :**

L'agent Utilisateur représente l'intermédiaire entre le patient et les autres agents du système. Il peut être considéré comme l'agent ayant le plus d'interaction dans notre modèle. Il a pour mission de : récolter les informations du patient, passer ces informations aux autres agents, afficher les résultats finaux au patient.

Cet agent récolte plusieurs informations auprès du patient dont l'image du fond d'œil qui est utilisée pour la classification, mais aussi des données statistiques relatives à la prédiction du risque d'apparition de la rétinopathie diabétique qu'il stocke dans son état interne et qui sont : Gender, age, Urea, Cr, HbA1c, Chol, TG, HDL, LDL, VLDL, BMI et un deuxième HbA1c. Ces données statistiques sont envoyées à l'agent prédiction en cas de besoin d'effectuer une prédiction ce qui veut dire dans le cas où la classe de l'image de fond d'œil est égale à « 0 ».

3.1.1.2 Agent classification

- **Rôle de l'agent :**

L'agent classification est un agent qui s'occupe de classer l'image de fond d'œil envoyée par l'agent utilisateur en l'une des cinq classes : 0 (Pour No DR donc pas de présence de rétinopathie diabétique), 1 (Pour mild non-proliférative DR donc rétinopathie diabétique non proliférante légère), 2 (Pour moderate non-proliférative DR donc rétinopathie diabétique non proliférante modérée), 3 (Pour sévère non-proliférative donc rétinopathie diabétique non proliférante sévère), 4 (Pour proliférative DR donc rétinopathie diabétique proliférante). Si la classe obtenue est la classe 0 alors l'agent envoie une notification à l'agent prédiction pour qu'il effectue une prédiction sinon il envoie le résultat obtenu à l'agent utilisateur pour l'afficher comme étant le résultat final.

- **Architecture interne de l'agent :**

L'agent classification est un réseau de neurones convolutif multi-classes. Il a comme entrée, l'image du fond d'œil du patient qu'il reçoit de l'agent utilisateur. Il a une sortie « softmax », tel qu'il y a 5 classes possibles décrites précédemment. Le résultat qui sera obtenu, sera la classe d'appartenance de l'image reçue.

Dans la figure 3.2, l'architecture interne de l'agent classification.

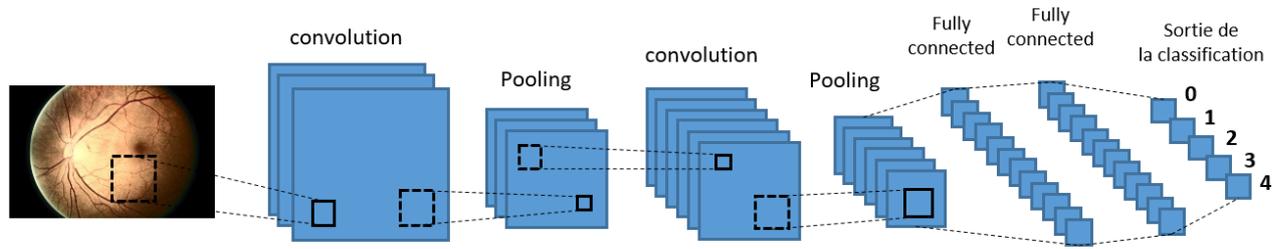


FIGURE 3.2 – Architecture interne de l’agent Classification

3.1.1.3 Agent prédiction

- **Rôle de l’agent :**

L’agent prédiction, comme son nom l’indique, s’occupe de prédire le risque de développement d’une rétinopathie diabétique chez un patient diabétique. Il agit sur des données appartenant au patient et reçues de l’agent utilisateur. À la fin ; Il retourne à l’agent utilisateur un résultat qui est soit « 1 » pour présence de risque d’apparition d’une RD chez le patient ou « 0 » pour absence de risque d’apparition d’une RD chez le patient.

- **Architecture interne de l’agent :**

L’agent prédiction est un réseau de neurones qui agit sur des données prédictives statistiques. Sa sortie est sigmoïde étant donné que c’est une prédiction binaire, nous n’avons que deux classes ; « 0 » ou « 1 » qui représentent le risque d’apparition d’une RD chez un patient diabétique. Ce réseau de neurone comporte 219701 paramètres et 4 couches : La première étant la couche d’entrée et comporte 12 neurones. La deuxième couche est la première couche cachée et comporte 700 neurones et a pour fonction d’activation, la fonction « ReLU ». La troisième couche est la deuxième couche cachée et comporte 300 neurones et a aussi pour fonction d’activation, la fonction « ReLU ». La fonction d’activation « ReLU » ou autrement dit Unité linéaire rectifiée est utilisée dans le but que toutes les valeurs négatives ne soient pas transmises à la couche suivante. La dernière couche est la couche de sortie et elle a pour fonction d’activation la fonction « Sigmoid » car notre modèle s’occupe de spécifier que deux labels « 0 » et « 1 ».

Dans la figure 3.3, l’architecture interne de l’agent prédiction.

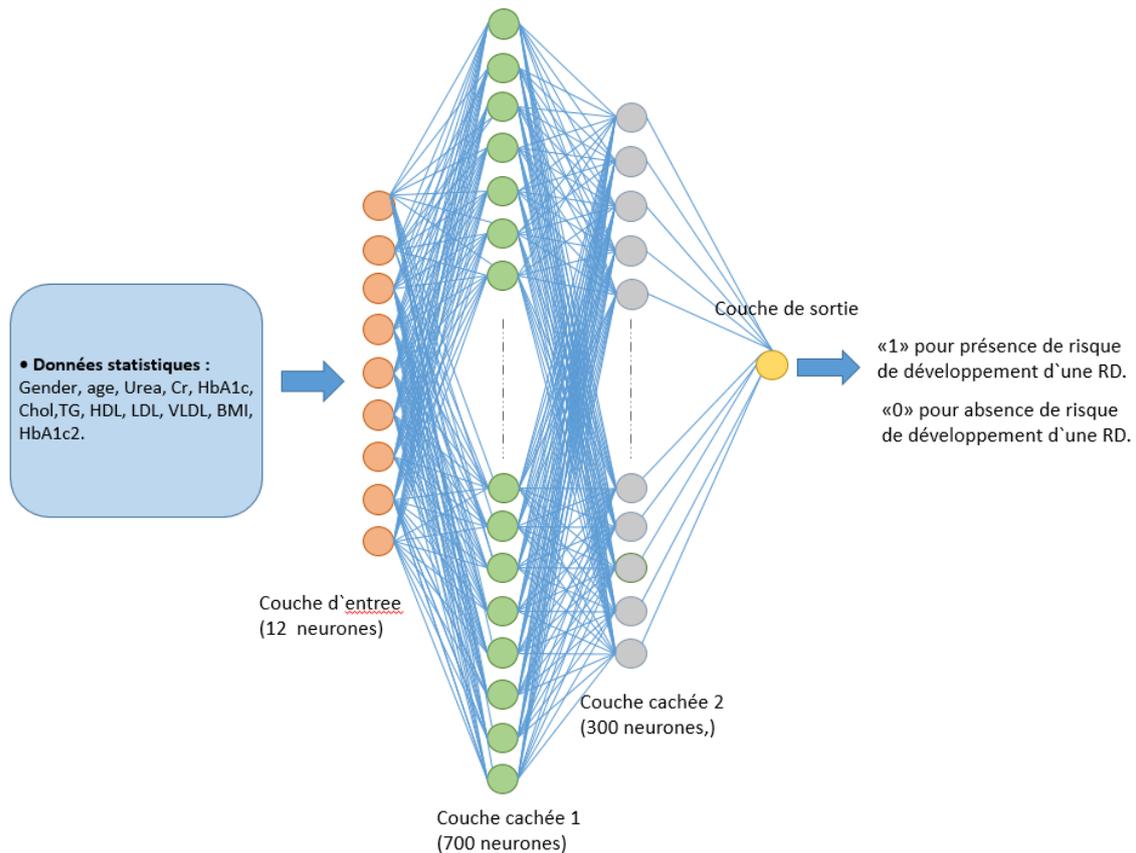


FIGURE 3.3 – Architecture interne de l'agent prédiction

3.1.2 Dynamique du système multi-agents proposé

Les différentes étapes du processus du système multi-agents proposé sont décrites dans ce qui suit :

- L'agent utilisateur récupère les données de l'utilisateur dont l'image de fond d'œil ainsi que des données statiques.
- L'agent utilisateur envoie l'image de fond d'œil à l'agent classification.
- L'agent classification réceptionne l'image de fond d'œil et effectue la classification :
Si le résultat de la classification est 1 ou 2 ou 3 ou 4 alors :
 - Le résultat est envoyé à l'agent utilisateur.
 - L'agent utilisateur reçoit le résultat et l'affiche à l'utilisateur.

Si le résultat de la classification est 0 alors :

- L'agent classification envoie un message à l'agent prédiction.
- L'agent prédiction réceptionne le message et envoie une demande à l'agent utilisateur pour lui envoyer le reste des données statiques récoltées auprès de l'utilisateur.
- L'agent utilisateur réceptionne la demande et envoie les données statiques relatives à la prédiction.
- L'agent prédiction réceptionne les données et effectue la prédiction.

- Une fois la prédiction effectuée, l'agent prédiction envoie les résultats finaux à l'agent utilisateur.
- L'agent utilisateur reçoit les résultats et les affiche à l'utilisateur.

La figure 3.4 représente un diagramme d'interactions entre les agents du SMA proposé.

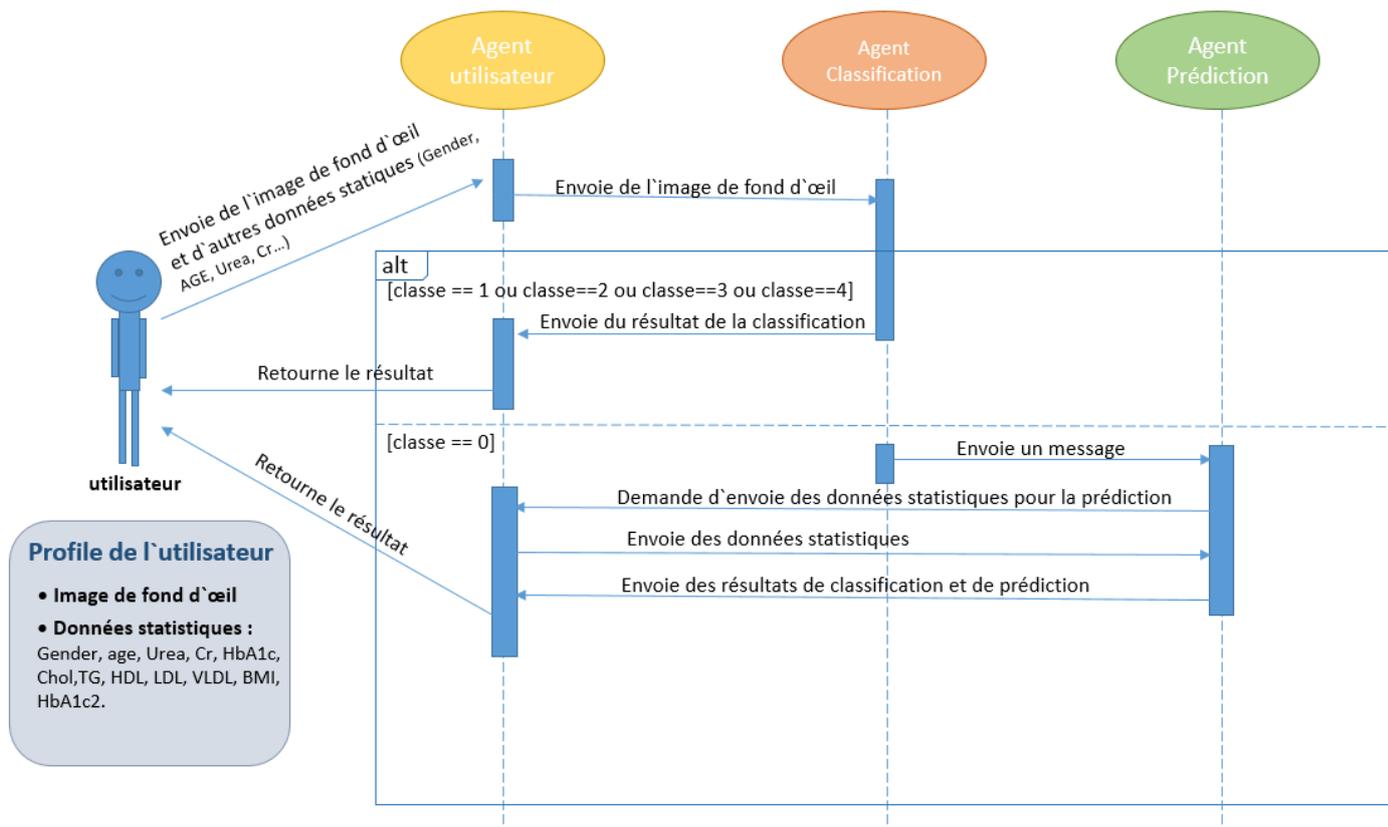


FIGURE 3.4 – Diagramme d'interactions du système multi-agents

Conclusion

Dans ce chapitre, nous avons présenté le système multi-agents que nous proposons pour la prédiction du risque de développement d'une rétinopathie diabétique chez un patient souffrant de diabète. Nous avons détaillé l'architecture de notre système multi-agents, ainsi que chaque agent de ce système. Et pour finir nous avons décrit la dynamique du système multi-agents proposé.

Le chapitre suivant sera consacré à la partie expérimentale où nous donneront suite à la réalisation de notre modélisation et aux résultats obtenus.

Chapitre 4

Expérimentation

Introduction

Dans le chapitre précédent nous avons décrit notre proposition de façon détaillée. Dans ce chapitre, nous allons présenter l'implémentation du modèle proposé ainsi que quelques scénarios d'exécution. Néanmoins nous nous sommes limitées à l'implémentation de la prédiction en considérant que l'image est déjà classée et que la classe est « 0 ».

4.1 outils et Bibliothèques utilisés

Dans ce qui suit, nous allons définir les bibliothèques les plus importantes pour la réalisation de notre modèle.

4.1.1 Anaconda

Anaconda est une distribution gratuite et open source des langages de programmation Python et R pour les applications liées à la science des données et à l'apprentissage automatique. Elle peut être installée sur Windows, Linux et MacOS. Le principal avantage de l'utilisation d'anaconda est qu'elle est comme un point central pour les bibliothèques dont on a besoin pour le traitement des données, l'analyse prédictive et les calculs scientifiques [51].

4.1.1.1 Spyder

Spyder est un environnement de développement gratuit et open source écrit en Python, pour Python. Il a été conçu par et pour des scientifiques, des ingénieurs et des analystes de données. Cet environnement de développement offre une combinaison unique des fonctionnalités avancées d'édition, d'analyse, de débogage et de profilage d'un outil de développement complet avec l'exploration de données, l'exécution interactive, l'inspection approfondie et les magnifiques capacités de visualisation d'un progiciel scientifique[52].

4.1.2 Python

Python est un langage de programmation interprété, multi-paradigme et multiplateformes. Il est placé sous une licence libre et fonctionne sur la plupart des plates-formes informatiques,

de Windows à Unix passant par macOS, ou encore Android, iOS. Python est actuellement le langage de programmation qui connaît la croissance la plus rapide au monde, grâce à sa facilité d'utilisation, sa courbe d'apprentissage rapide et ses nombreux packages de haute qualité pour la science des données et l'apprentissage automatique [53].

Ce langage comporte plusieurs bibliothèques dont celles que nous avons utilisées :

4.1.2.1 SPADE

SPADE (Smart Python Agent Development Environment) est une plateforme de systèmes multi-agents écrite en Python et basée sur la messagerie instantanée « XMPP ». Grâce à cette plateforme, il est possible de développer des agents capables d'interagir à la fois avec d'autres agents et avec des humains[54].

4.1.2.2 Matplotlib

Matplotlib est une bibliothèque complète permettant de créer des visualisations statiques, animées et interactives en Python[55].

4.1.2.3 Pandas

Pandas est une bibliothèque Python utilisée pour la manipulation et l'analyse des données. Elle fournit des structures de données rapides, flexibles et expressives conçues pour rendre le travail avec des données "relationnelles" ou "étiquetées" à la fois facile et intuitif[56].

4.1.2.4 Numpy

NumPy est une bibliothèque Python utilisée pour manipuler des matrices ou tableaux multidimensionnels ainsi que des fonctions mathématiques qui opèrent sur ces tableaux.

4.1.2.5 Scikit-learn

Scikit-learn est une bibliothèque libre Python destinée à l'apprentissage automatique. Cette bibliothèque contient de nombreux algorithmes dont les forêts aléatoires, la régression logistique, des algorithmes de classification. . .

4.1.2.6 Scikit-plot

Une bibliothèque intuitive pour ajouter une fonctionnalité de traçage aux objets scikit-learn. Elle peut vous aider à visualiser vos données, votre modèle (pendant la formation) et les résultats de vos expériences à différentes étapes de votre projet d'apprentissage automatique [57].

4.1.2.7 Tensorflow

Tensorflow est une bibliothèque python open source d'apprentissage profond, permettant de définir, d'entraîner et de déployer des modèles d'apprentissage automatique[58].

4.1.2.7.1 Keras : Keras est une bibliothèque Python de haut niveau, compacte et facile à apprendre, pour l'apprentissage profond. TensorFlow étant considéré comme étant le back-end pour Keras. Cette bibliothèque permet aux développeurs de se concentrer sur les principaux concepts de l'apprentissage profond, comme la création de couches pour les réseaux neuronaux, tout en s'occupant des détails minutieux des tenseurs, de leurs formes et de leurs détails mathématiques[59].

4.2 Définition de l'ensemble des données utilisé et description des variables

Afin de réaliser l'évaluation de notre étude, nous avons eu besoin d'un jeu de données. Dans cette section nous définirons l'ensemble des données utilisées ainsi que les variables le composant.

4.2.1 Définition de l'ensemble des données utilisé

Afin de pouvoir valider notre approche nous avons été contraint d'adapter le dataset [60] et ce dû au manque et à l'incomplétude des informations pour la réalisation de la prédiction de l'apparition de la rétinopathie diabétique.

L'ensemble des données [60] a été collecté auprès de la société irakienne. En effet, ces données ont été acquises auprès du laboratoire du Medical City Hospital ainsi qu'auprès du Specializes Center for Endocrinology and Diabetes-Al-Kindy Teaching Hospital. Elle contenait plusieurs variables que nous décrirons dans la prochaine section. Dans notre cas nous avons dû ajouter une colonne pour le HbA1c2 et changer la colonne classe pour qu'elle représente le risque d'apparition de la RD chez le patient. Pour cela nous avons suivie l'étude de A. Larroumet[61]. Il a été montré dans cette étude qu'une différence de 3% entre deux HbA1c consécutifs représentait un risque majeur de l'apparition de la RD. Nous avons donc pris en considération ces informations afin d'élaborer les colonnes Hb1Ac2 et la colonne classe suivant les différents cas possibles pour la prédiction.

La figure 4.1 représente un aperçu de la base de données que nous avons élaborée pour notre étude.

	No_Pation	Gender	AGE	Urea	Cr	HbA1c	...	HDL	LDL	VLDL	BMI	HbA1c2	Class
0	23975	M	31	3.0	60	12.3	...	0.7	2.4	15.4	37.2	7.9	1
1	23977	M	30	7.1	81	6.7	...	1.2	2.4	8.1	27.4	6.5	0
2	23979	M	45	4.1	63	10.2	...	0.9	3.3	9.5	34.3	9.2	0
3	33656789	M	45	4.1	63	10.2	...	0.9	3.3	9.5	34.3	13.3	1
4	23984	M	45	5.3	77	11.2	...	1.3	2.0	10.4	29.5	11.0	0
5	67036	M	31	3.4	55	6.5	...	1.0	3.2	0.7	24.0	6.5	0
6	745326	M	45	5.3	77	11.2	...	1.3	2.0	10.4	29.5	5.0	1
7	23987	M	30	5.0	80	6.9	...	1.2	2.6	12.7	34.6	11.4	1
8	768585	M	30	5.0	80	6.9	...	1.2	2.6	12.7	34.6	5.8	0
9	23990	M	35	4.8	64	7.7	...	1.2	2.0	7.2	27.3	7.6	0

FIGURE 4.1 – Aperçu du dataset adapté

Dans la prochaine section, nous définirons les variables de la base de données adaptée pour la prédiction de la rétinopathie diabétique.

4.2.2 Description des variables

La base de données que nous avons obtenue après adaptation est composée de plusieurs variables prédictives médicales et d'une variable cible « Class » :

Variable	Description
Gender	Sexe d'une personne soit Féminin soit Masculin.
AGE	Age d'une personne.
Urea	L'urée est un produit de la dégradation des protéines qui est éliminé par les urines. Les taux d'urée dans le sang sont donc un reflet de la fonction rénale, et, dans certaines conditions, de l'apport alimentaire en protéines ainsi que du fonctionnement du foie.
Cr	le taux de créatinine. La concentration normale de la créatinine dans le sang est comprise entre 6 et 12 mg/l (milligrammes par litre) chez l'homme et entre 4 et 10 mg/l chez la femme.
HbA1c	L'HbA1c (hémoglobine glyquée) est le pourcentage d'hémoglobine ayant fixé du sucre dans le sang. Elle est le reflet de la glycémie moyenne (taux de sucre dans le sang) des 3 derniers mois précédant le dosage en laboratoire.
Chol	Le cholestérol est un type de lipide (gras) fabriqué par le foie et présent dans les aliments que l'on consomme. Le taux normal de cholestérol total est ($< 5,0$ mmol/L).
TG	les triglycérides sont, comme le cholestérol, des lipides composés de molécules de glycérol et d'acides gras et conservés dans les tissus adipeux du corps. Le taux normal de triglycérides est ($< 1,7$ mmol/L).
HDL	Les lipoprotéines HDL ou autrement dit le bon cholestérol captent le cholestérol qui se dépose dans les artères et le transportent vers le foie où il est éliminé. Le taux normal de cholestérol HDL est ($> 1,0$ mmol/L pour les hommes, $> 1,3$ mmol/L pour les femmes).

LDL	Les lipoprotéines LDL ou autrement dit le mauvais cholestérol est néfastes à l'organisme, fixent le cholestérol sur les parois des artères et entraînent la formation de plaques d'athérome. Le taux normal de cholestérol LDL est ($< 3,5$ mmol/L).
VLDL	Lipoprotéine de très basse densité est une classe de lipoprotéines plasmatiques très riches en triglycérides. Elles servent à transporter le cholestérol qui a été synthétisé par le foie. Le taux idéal de VLDL est quand il est au-dessous de 0,3 g/L.
BMI	(Poids en kg / taille en m ²) IMC de 18.5 à 20 c'est normal IMC entre 25 et 30 situer dans une plage surpoids Et de 30 ou plus situer dans la fourchette d'obésité.
HbA1c2	Ceci représente le HbA1c consecutif du HbA1c de la colonne précédente.
Class	Indique si une personne est en risque d'avoir une rétinopathie diabétique.

4.3 Implémentation de l'agent Prédiction

Afin d'effectuer la prédiction, nous avons opté pour deux approches la première étant d'utiliser un classificateur Random Forest et la deuxième étant d'utiliser le Deep Learning. Nous les avons comparés en terme de résultats afin de choisir quelle approche utilisée dans notre système multi-agents.

4.3.1 Utilisation de la méthode Random Forest

Une technique faisant partie des techniques de l'apprentissage automatique, Random Forest est utilisé pour la classification mais aussi pour la prédiction ce qui est notre cas dans cette étude. Cette technique est composée de plusieurs arbres décisionnels. La forêt d'arbres décisionnels que nous avons implémentée comporte 440 arbres de décisions.

4.3.1.1 Implémentation

Nous avons d'abord importé les bibliothèques dont nous avons besoin pour effectuer le Random Forest. Dans la figure 4.2 , les bibliothèques importées.

```

from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report
from sklearn.metrics import confusion_matrix
import numpy as np
import pandas as pd
import scikitplot as skplot
import matplotlib.pyplot as plt

```

FIGURE 4.2 – Bibliothèques importées pour Random Forest

Nous avons ensuite importé la base de données. La colonne « N_Pation » a été supprimée et les valeurs de la colonne « Gender » ont été remplacées; « 0 » pour « F » et « 1 » pour « M ».

```

dataset=pd.read_csv("dataset.csv")

dataset.drop(["No_Pation"],axis="columns",inplace=True)

dataset["Gender"].replace(["F","f","M"],[0,0,1],inplace=True)

```

FIGURE 4.3 – Importation du dataset et modifications

Suite à ces modifications nous avons obtenu ce qui suit :

	Gender	AGE	Urea	Cr	HbA1c	Chol	TG	HDL	LDL	VLDL	BMI	HbA1c2	Class
0	1	31	3.0	60	12.3	4.1	2.2	0.7	2.4	15.4	37.2	7.9	1
1	1	30	7.1	81	6.7	4.1	1.1	1.2	2.4	8.1	27.4	6.5	0
2	1	45	4.1	63	10.2	4.8	1.3	0.9	3.3	9.5	34.3	9.2	0
3	1	45	4.1	63	10.2	4.8	1.3	0.9	3.3	9.5	34.3	13.3	1
4	1	45	5.3	77	11.2	3.9	1.5	1.3	2.0	10.4	29.5	11.0	0
5	1	31	3.4	55	6.5	4.9	1.6	1.0	3.2	0.7	24.0	6.5	0
6	1	45	5.3	77	11.2	3.9	1.5	1.3	2.0	10.4	29.5	5.0	1
7	1	30	5.0	80	6.9	4.5	1.8	1.2	2.6	12.7	34.6	11.4	1
8	1	30	5.0	80	6.9	4.5	1.8	1.2	2.6	12.7	34.6	5.8	0
9	1	35	4.8	64	7.7	3.7	1.0	1.2	2.0	7.2	27.3	7.6	0

FIGURE 4.4 – Aperçu après modification

Le jeu de données a ensuite été séparé en données d'entraînement et en données testes. Les données de testes représentent 20% du jeu de données, tandis que les données d'entraînement représentent 80%.

```
#x represente toutes les colonnes sauf la derniere
X=dataset.iloc[:, :-1]

#y represente la derniere colonne seulement
y=dataset.iloc[:, [-1]]

#les donnees test 20%, les donnees d'entrainement 80%
x_train,x_test,y_train,y_test=train_test_split(X,y,test_size=.2, random_state=0)
```

FIGURE 4.5 – Séparation des données en données testées et entrainement

Notre classificateur Random Forest a été créé et les données ont été envoyées au classificateur.

```
classifieur=RandomForestClassifier(n_estimators=440, criterion="entropy", random_state=1)
classifieur.fit(x_train,y_train)
```

FIGURE 4.6 – Random Forest

- `n_estimators` : représente le nombre d'arbre dans la forêt dans notre cas nous avons 440 arbres de décisions.
- `criterion` : est un paramètre spécifique à l'arbre. C'est une fonction qui permet de mesurer la qualité d'un fractionnement.
- `random_state` Contrôle à la fois le caractère aléatoire du bootstrap des échantillons utilisés lors de la construction des arbres et l'échantillonnage des caractéristiques à prendre en compte lors de la recherche de la meilleure division à chaque nœud.

4.3.1.2 Résultats obtenus avec Random Forest

Grâce à Random Forest nous avons obtenu une accuracy de 85.20%, ce qui représente un score très encourageant.

Python permet, grâce à certaines de ces fonctions d'obtenir l'accuracy mais aussi d'obtenir un rapport de classification. Pour obtenir l'accuracy nous avons utilisé l'instruction suivante :

```
print("l'accuracy est: ", classifieur.score(x_test, y_test))
```

FIGURE 4.7 – Instruction pour l'affichage de l'accuracy de Random Forest

Pour afficher le rapport de classification nous utilisons la fonction `classification_report()` de `sklearn.metrics`. La figure ci-après montre l'instruction Python dédiée à cela : Pour afficher le rapport de classification nous utilisons la fonction `classification_report()` de `sklearn.metrics`. La figure ci-après montre l'instruction Python dédiée à cela :

```
print(classification_report(y_test, y_pred))
```

FIGURE 4.8 – Instruction pour l’affichage du rapport de classification de Random Forest

Nous obtenons ce qui suit grâce à l’instruction qui affiche l’accuracy et celle qui affiche le rapport de classification. L’accuracy obtenue est de 0.852 soit de 85.20%. Le rapport de classification montre la precision, le recall et le f1-score ainsi que l’accuracy.

```
l`accuracy est: 0.8520710059171598
      precision    recall  f1-score   support

0     0.84     0.89     0.86         87
1     0.87     0.82     0.84         82

 accuracy
macro avg    0.85     0.85     0.85        169
weighted avg 0.85     0.85     0.85        169
```

FIGURE 4.9 – Rapport de classification de Random Forest

Nous avons aussi procédé à l’affichage de la matrice de confusion et ce comme suit :

```
y_pred=classifieur.predict(x_test)
cm=confusion_matrix(y_test, y_pred)
print(cm)

print("l`accuracy est: ", classifieur.score(x_test, y_test))

skplot.metrics.plot_confusion_matrix(y_test,y_pred,normalize=False)
plt.show()
```

FIGURE 4.10 – Instruction pour l’affichage de la matrice de confusion de Random Forest

La matrice de confusion est la suivante :

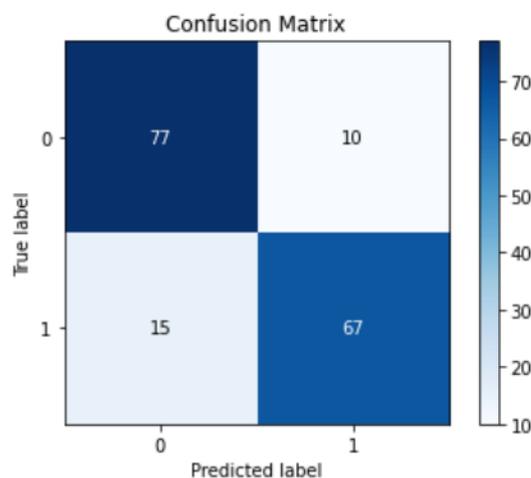


FIGURE 4.11 – Matrice de confusion de Random Forest

Les résultats obtenus avec cette approche sont très intéressants en terme d'exactitude.

4.3.1.3 Scénario d'exécution

Nous avons voulu évaluer le Random Forest sur des données appartenant à certains patients diabétiques. Ces informations ont été récoltées lors de notre stage au CHU de Bejaia.

Nous avons testé le modèle sur trois profils de trois patients différents. Nous considérons, bien sûr, que leurs images de fond d'œil ont déjà été classées et que la classe obtenue a été la classe « 0 » donc absence de rétinopathie diabétique pour chacun des profils.

Dans ce qui suit les profils des trois patients :

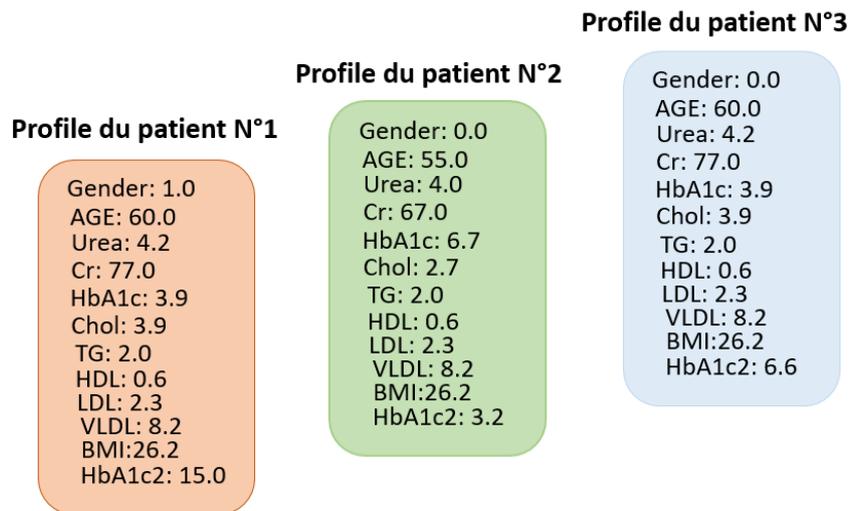


FIGURE 4.12 – Profiles de trois patients pour tester le processus de prédictions

Afin d'obtenir le résultat de la prédiction du risque de développement de la RD chez le patient diabétique N°1, on utilise l'instruction illustrée dans la figure 4.13.

```
print("la nouvelle prediction est: ",classfier.predict(np.array([[1.0,60.0,4.2,77.0,3.9,3.9,2.0,0.6,2.3,8.2,26.2,15.0]]))))
```

FIGURE 4.13 – Instruction pour prédire le risque d'apparition de la rétinopathie diabétique chez le patient N°1

Le résultat de la prédiction pour le patient N°1 est « 1 » donc il y a un risque d'apparition d'une RD chez ce patient. Le résultat est illustré ci-joint, dans la figure 4.14.

```
la nouvelle prediction est: [1]
```

FIGURE 4.14 – Résultat de la prédiction du risque d'apparition de la rétinopathie diabétique chez le patient N°1

Pour la prédiction du risque de développement de la RD chez le patient diabétique N°2, on utilise l'instruction de la figure 4.15.

```
print("la nouvelle prediction est: ",classifler.predict(np.array([[0.0,55.0,4.0,67.0,6.7,2.7,2.0,0.6,2.3,8.2,26.2,3.2]])))
```

FIGURE 4.15 – Instruction pour prédire le risque d’apparition de la rétinopathie diabétique chez le patient N°2

Le résultat de la prédiction pour le patient N°2 est « 1 » donc il y a un risque d’apparition d’une RD chez ce patient. Le résultat est illustré ci-joint, dans la figure 4.16.

```
la nouvelle prediction est: [1]
```

FIGURE 4.16 – Résultat de la prédiction du risque d’apparition de la rétinopathie diabétique chez le patient N°2

Pour la prédiction du risque de développement de la RD chez le patient diabétique N°3, on utilise l’instruction de la figure 4.17.

```
print("la nouvelle prediction est: ",classifler.predict(np.array([[0.0,55.0,4.0,67.0,6.7,2.7,2.0,0.6,2.3,8.2,26.2,6.6]])))
```

FIGURE 4.17 – Instruction pour prédire le risque d’apparition de la rétinopathie diabétique chez le patient N°3

Le résultat de la prédiction pour le patient N°3 est « 0 » donc il y a un risque d’apparition d’une RD chez ce patient. Le résultat est illustré ci-joint, dans la figure 4.18.

```
la nouvelle prediction est: [0]
```

FIGURE 4.18 – Résultat de la prédiction du risque d’apparition de la rétinopathie diabétique chez le patient N°3

Les résultats obtenus ont été ceux auxquels on s’attendait et donc ceci représente un bon présage en ce qui concerne cette démarche.

4.3.2 Utilisation du Deep Learning

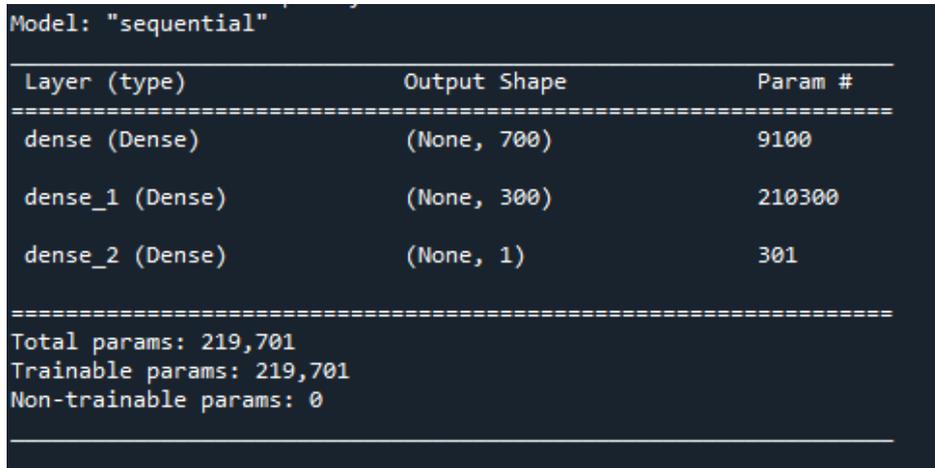
Nous proposons dans cette approche un réseau de neurones pour la prédiction du risque d’apparition de la rétinopathie diabétique. Nous allons décrire l’architecture de ce réseau de neurones puis nous donnerons les étapes de l’implémentation du modèle et pour finir nous parlerons des résultats obtenus avec ce dernier.

4.3.2.1 Architecture du réseau de neurones

Nous avons opté pour un réseau de neurones assez simple. Ce réseau se compose de 4 couches seulement dont une couche d’entrée, deux cachées et une de sortie. La première couche qui est la couche d’entrée comporte 12 neurones tandis que la première couche cachée en comporte 700 neurones, la deuxième couche cachée en comporte 300 et celle de sortie 1 neurone. Ce réseau

comporte 219701 paramètres.

La figure 4.19 résume l'architecture du modèle que nous avons implémenté sur l'environnement Spyder. Pour obtenir ce résumé il faut seulement utiliser l'instruction `model.summary()`.



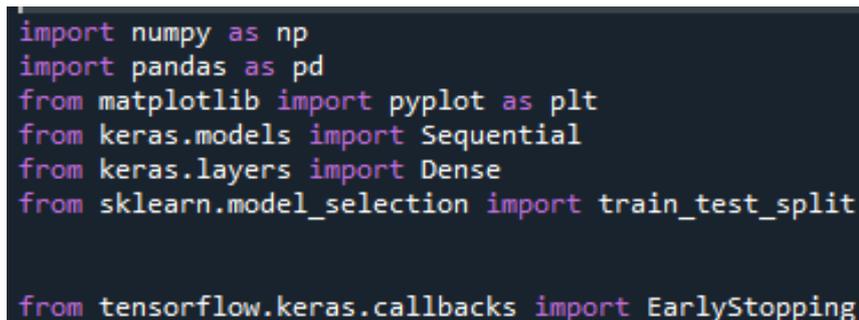
```
Model: "sequential"
-----
Layer (type)                 Output Shape                 Param #
-----
dense (Dense)                 (None, 700)                 9100
dense_1 (Dense)               (None, 300)                210300
dense_2 (Dense)               (None, 1)                   301
-----
Total params: 219,701
Trainable params: 219,701
Non-trainable params: 0
-----
```

FIGURE 4.19 – Résumé du modèle

4.3.2.2 Implémentation

Les étapes d'importation du dataset et de modifications de ce dernier sont les mêmes que celles effectuées pour le Random Forest. Cela dit, il y a une différence dans les bibliothèques importées pour la réussite de l'implémentation.

La figure 4.20 montre les différentes bibliothèques à télécharger pour mener à bien l'implémentation de notre modèle.



```
import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
from keras.models import Sequential
from keras.layers import Dense
from sklearn.model_selection import train_test_split

from tensorflow.keras.callbacks import EarlyStopping
```

FIGURE 4.20 – Bibliothèques importées pour le Deep Learning

Nous avons ensuite importé la base de données et effectué les modifications requises comme pour le Random Forest. Puis nous avons aussi divisé le jeu de données en données de tests et données d'entraînement comme le montre la figure 4.21.

```

dataframeInNumpy=dataset.values
inputVariables=dataframeInNumpy[:, :12]
outputVariables=dataframeInNumpy[:, 12:]

x_train, x_test,y_train,y_test= train_test_split(inputVariables, outputVariables, test_size=.2, random_state=42)

```

FIGURE 4.21 – Division du jeu de données en données de tests et données d’entraînement pour le Deep Learning

La création du réseau de neurones a été réalisée en utilisant la bibliothèque Keras. Le modèle étant un modèle simple basée sur une couche entrée, 2 couches cachées et une couche de sortie ; la première couche qui est la couche entrée contient 12 neurones, tandis que la première couche cachée en contient 700 et la deuxième couche cachée est composée de 300 neurones. Pour ce qui est de la dernière couche du réseau, elle contient 1 neurone et représente la couche de sortie. En ce qui s’agit des fonctions d’activation, les deux couches cachées avaient pour fonction d’activation la fonction « ReLU », et pour la dernière il s’agissait de la fonction d’activation « Sigmoid » étant donné qu’elle se trouve être la fonction la plus adéquate pour notre cas. En effet, la fonction « Sigmoid » reste la plus adaptée quand il s’agit d’un modèle qui doit déterminer seulement deux labels, « 0 » ou « 1 » dans notre cas. La figure 4.22 montre les instructions suivies pour la création de notre réseau de neurones et sa compilation.

```

model=Sequential()
model.add(Dense(700,input_dim=12 ,activation="relu"))
model.add(Dense(300,activation="relu"))
model.add(Dense(1,activation="sigmoid"))

model.compile(loss="binary_crossentropy",optimizer="adam", metrics=["accuracy"])

```

FIGURE 4.22 – Instructions pour la création de notre réseau de neurones et sa compilation

Notre modèle souffre de sur-apprentissage alors nous avons utilisé la méthode EarlyStopping. Cette méthode arrête la formation lorsqu’une mesure surveillée a cessé de s’améliorer, dans notre cas la mesure surveillée est le « val_loss » étant donné que le but de l’entraînement est de minimiser ce dernier.

Dans la figure 4.23 , l’instruction permettant d’appliquer le EarlyStopping a notre modèle pour tenter de régler le problème de sur-apprentissage.

```

callback=EarlyStopping(
    monitor="val_loss",
    min_delta=0.00001,
    patience=20,
    verbose=1,
    mode="auto",
    baseline=None,
    restore_best_weights=False)

history=model.fit(x_train,y_train,epochs=300,batch_size=70, validation_data=(x_test,y_test), callbacks=callback)

```

FIGURE 4.23 – Application du EarlyStopping

4.3.2.3 Résultats obtenus avec l'approche Deep Learning

Grâce au modèle de Deep Learning développé, nous avons obtenu une accuracy de 96.92%, ce qui représente un très bon résultat .

Python permet d'obtenir l'accuracy en utilisant les instructions suivantes :

```

scores=model.evaluate(inputVariables,outputVariables)

print("\n%s: %.2f%%" %(model.metrics_names[1], scores[1]*100))

```

FIGURE 4.24 – Instructions pour obtenir l'accuracy du modèle de Deep Learning développé

Les résultats obtenus sont décrits dans la figure 4.25.

```

Epoch 132/300
10/10 [=====] - 0s 7ms/step - loss: 0.1341 - accuracy: 0.9452 - val_loss: 0.2269 -
val_accuracy: 0.9231
Epoch 133/300
10/10 [=====] - 0s 7ms/step - loss: 0.1142 - accuracy: 0.9600 - val_loss: 0.2313 -
val_accuracy: 0.9231
Epoch 134/300
10/10 [=====] - 0s 7ms/step - loss: 0.1030 - accuracy: 0.9630 - val_loss: 0.2614 -
val_accuracy: 0.9349
Epoch 135/300
10/10 [=====] - 0s 7ms/step - loss: 0.1011 - accuracy: 0.9674 - val_loss: 0.2373 -
val_accuracy: 0.9349
Epoch 135: early stopping
27/27 [=====] - 0s 2ms/step - loss: 0.1173 - accuracy: 0.9692

accuracy: 96.92%

```

FIGURE 4.25 – Résultats obtenus avec le modèle de Deep Learning développé

4.3.2.4 Scénario d'exécution

Comme pour le Random Forest, nous avons testé avec les mêmes profils de patients décrit dans la figure 4.12.

Afin d'obtenir le résultat de la prédiction du risque de développement de la RD chez le patient diabétique N°1, on utilise l'instruction illustrée dans la figure 4.26.

```
print("La nouvelle prediction est: ",model.predict(np.array([[1.0,60.0,4.2,77.0,3.9,3.9,2.0,0.6,2.3,8.2,26.2,15.0]])))
```

FIGURE 4.26 – Instruction pour prédire le risque d'apparition de la rétinopathie diabétique chez le patient N°1 avec Deep Learning

Le résultat de la prédiction pour le patient N°1 est « 1 » donc il y a un risque d'apparition d'une RD chez ce patient. Le résultat est illustré ci-joint, dans la figure 4.27.

```
1/1 [=====] - 0s 62ms/step
la nouvelle prediction est: [[1.]]
```

FIGURE 4.27 – Résultat de la prédiction du risque d'apparition de la rétinopathie diabétique chez le patient N°1 en utilisant le Deep Learning

Pour la prédiction du risque de développement de la RD chez le patient diabétique N°2, on utilise l'instruction de la figure 4.28.

```
print("La nouvelle prediction est: ",model.predict(np.array([[0.0,55.0,4.0,67.0,6.7,2.7,2.0,0.6,2.3,8.2,26.2,3.2]])))
```

FIGURE 4.28 – Instruction pour prédire le risque d'apparition de la rétinopathie diabétique chez le patient N°2 en utilisant le Deep Learning

Le résultat de la prédiction pour le patient N°2 est « 1 » donc il y a un risque d'apparition d'une RD chez ce patient. Le résultat est illustré ci-joint, dans la figure 4.29.

```
1/1 [=====] - 0s 16ms/step
la nouvelle prediction est: [[0.98015815]]
```

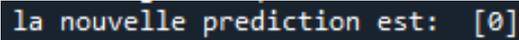
FIGURE 4.29 – Résultat de la prédiction du risque d'apparition de la rétinopathie diabétique chez le patient N°2 en utilisant le Deep Learning

Pour la prédiction du risque de développement de la RD chez le patient diabétique N°3, on utilise l'instruction de la figure 4.30.

```
print("La nouvelle prediction est: ",model.predict(np.array([[0.0,55.0,4.0,67.0,6.7,2.7,2.0,0.6,2.3,8.2,26.2,6.6]])))
```

FIGURE 4.30 – Instruction pour prédire le risque d'apparition de la rétinopathie diabétique chez le patient N°3

Le résultat de la prédiction pour le patient N°3 est « 0 » donc il y a un risque d'apparition d'une RD chez ce patient. Le résultat est illustré ci-joint, dans la figure 4.31.



```
la nouvelle prediction est: [0]
```

FIGURE 4.31 – Résultat de la prédiction du risque d’apparition de la rétinopathie diabétique chez le patient N°3 avec Deep Learning

4.4 Discussion des résultats

Comme vu dans la section précédentes les deux approches ont été évaluées et des résultats ont été mis à l’évidence. Ces résultats étaient très encourageant, néanmoins, l’approche du Deep Learning dépasse le Random Forest.

La métrique utilisée pour juger que Deep Learning est meilleure est :

Accuracy : ou exactitude es le rapport : $(VP+VN)/(VP+VN+FP+FN)$. Plus proche de 1 meilleur est le teste.

Les résultats sont décrits dans le tableau ci-dessous :

	Random Forest	Deep Learning
Accuracy	85.20%	96.92%

Etant donné que le métrique d’évaluation sélectionnée est l’exactitude et que le modèle de Deep Learning obtient une accuracy nettement supérieure à celle obtenue par le Random Forest, nous avons donc décidé de mettre le modèle de Deep Learning dans l’agent perdition.

Conclusion

Dans ce chapitre, nous avons défini les bibliothèques et outils utilisés pour l’implémentation, nous avons décrit l’ensemble des données utilisées ainsi que les variables le composant. Cela dit, en terme d’implémentation nous nous sommes limitées à l’implémentation de la prédiction du risque de développement d’une rétinopathie diabétique chez un patient atteint de diabète. Nous avons développé deux modèles ; un modèle de Random Forest ainsi qu’un modèle basé Deep Learning. Nous les avons comparés par rapport aux résultats obtenue et au final nous avons décidé d’intégrer le modèle basé Deep Learning dans notre agent Prédiction car ses résultats étaient supérieures a ceux du Random Forest.

Conclusion Générale

La rétinopathie diabétique est une maladie pouvant mener à la cécité et donc à la perte totale de la vision humaine. Plusieurs chercheurs se sont intéressés à cette pathologie que ce soit pour sa détection, sa classification ou encore sa prédiction. L'informatique et en particulier l'intelligence artificielle ne cesse de faire ses preuves dans le domaine de la santé et du médical. C'est une discipline, qui grâce à ses nombreuses techniques permet d'effectuer des diagnostics médicaux, de la prédiction et bien d'autre.

Dans notre étude, nous avons proposé un modèle multi-agents basé sur l'apprentissage automatique, qui de par sa conception est unique d'après nos connaissances et recherches que nous avons menées. C'est un système modulaire, robuste et évolutif où chaque module qui est un agent est responsable d'une tâche spécifique et communique avec le reste des agents pour mener à bien sa mission. Ce modèle, grâce à l'interaction et à la communication des agents qui le composent permet d'effectuer une détection et classification de la rétinopathie diabétique mais aussi une prédiction du risque d'apparition de cette maladie chez un diabétique non-atteint de rétinopathie. En effet, dans le cas où le patient n'a pas de RD notre système effectue automatiquement une prédiction du risque de développer cette pathologie. Ceci pourrait permettre au patient d'éviter de contracter cette maladie en surveillant les facteurs de risque la favorisant tel que le HbA1c.

Nous avons utilisé plusieurs environnements et outils de développement pour implémenter le modèle proposé. Cependant, par manque de temps, nous nous sommes limités à l'implémentation de la partie prédiction, qui a donné des résultats très encourageants en matière d'exactitude. Nous avons implémenté deux approches pour la prédiction la première étant le Random Forest (85.20% d'exactitude) et une approche basée Deep Learning (96.92 % d'exactitude).

Comme perspectives, nous envisageons de finaliser l'implémentation de notre modèle et effectuer les tests nécessaires. Nous proposons également d'améliorer notre modèle afin qu'il puisse prédire l'évolution de la RD.

Bibliographie

- [1] Harry Pratt, Frans Coenen, Deborah M. Broadbent, Simon P. Harding, and Yalin Zheng. Convolutional neural networks for diabetic retinopathy. *Procedia Computer Science*, 90 :200–205, 2016. 20th Conference on Medical Image Understanding and Analysis (MIUA 2016).
- [2] Suvajit Dutta, Bonthala Manideep, Muzamil Basha, Ronnie Caytiles, and N Ch Sriman Narayana Iyenger. Classification of diabetic retinopathy images by using deep learning models. *International Journal of Grid and Distributed Computing*, 11(01) :89–106, 2018. 10.14257/ijgdc.2018.11.1.09.
- [3] Xiaoliang Wang, Yongjin Lu, Yujuan Wang, and Wei-Bang Chen. Diabetic retinopathy stage classification using convolutional neural networks. In *2018 IEEE International Conference on Information Reuse and Integration (IRI)*, pages 465–471, 2018. 10.1109/IRI.2018.00074.
- [4] Balazs Harangi, Janos Toth, Agnes Baran, and Andras Hajdu. Automatic screening of fundus images using a combination of convolutional neural network and hand-crafted features. In *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 2699–2702, 2019. 10.1109/EMBC.2019.8857073.
- [5] Tao Li, Yingqi Gao, Kai Wang, Song Guo, Hanruo Liu, and Hong Kang. Diagnostic assessment of deep learning algorithms for diabetic retinopathy screening. *Information Sciences*, 501 :511–522, 2019. <https://doi.org/10.1016/j.ins.2019.06.011>.
- [6] T. Shanthi and R.S. Sabeenian. Modified alexnet architecture for classification of diabetic retinopathy images. *Computers Electrical Engineering*, 76 :56–64, 2019. <https://doi.org/10.1016/j.compeleceng.2019.03.004>.
- [7] Benjamin N. Rao, Nicola Quinn, Andrzej S. Januszewski, Tunde Peto, Laima Brazionis, Nanda Aryal, Rachel L. O’Connell, Liping Li, Paula Summanen, Russell Scott, Justin O’Day, Anthony C. Keech, and Alicia J. Jenkins. Retinopathy risk calculators in the prediction of sight-threatening diabetic retinopathy in type 2 diabetes : A field substudy. *Diabetes Research and Clinical Practice*, 186 :109835, 2022. <https://doi.org/10.1016/j.diabres.2022.109835>.
- [8] Lund SH, Aspelund T, Kirby P, Russell G, Einarsson S, Palsson O, and Stefánsson E. Individualised risk assessment for diabetic retinopathy and optimisation of screening intervals : a scientific approach to reducing healthcare costs. *Br J Ophthalmol.*, 100(5) :683 – 687, 2016. 10.1136/bjophthalmol-2015-307341.
- [9] Wanyue Li, Yanan Song, Kang Chen, Jun Ying, Zhong Zheng, Shen Qiao, Ming Yang, Maonian Zhang, and Ying Zhang. Predictive model and risk analysis for diabetic retino-

- pathy using machine learning : a retrospective cohort study in china. *BMJ Open*, 11(11), 2021. [10.1136/bmjopen-2021-050989](https://doi.org/10.1136/bmjopen-2021-050989).
- [10] Hsin-Yi Tsao, Pei-Ying Chan, and Emily Chia-Yu Su. Predicting diabetic retinopathy and identifying interpretable biomedical features using machine learning algorithms. *BMC Bioinformatics*, 19(283), 2018. <https://doi.org/10.1186/s12859-018-2277-0>.
- [11] Ein Oh, Tae Keun Yoo, and Eun-Cheol Park. Diabetic retinopathy risk prediction for fundus examination using sparse learning : a cross-sectional study. *BMC Medical Informatics and Decision Making*, 13(106), 2013. <https://doi.org/10.1186/1472-6947-13-106>.
- [12] Gaurav Saxena, Dharendra Kumar Verma, Amit Paraye, Alpna Rajan, and Anil Rawat. Improved and robust deep learning agent for preliminary detection of diabetic retinopathy using public datasets. *Intelligence-Based Medicine*, 3-4 :100022, 2020. <https://doi.org/10.1016/j.ibmed.2020.100022>.
- [13] Hanane Alloui, Mohamed Sadgal, and Aziz El Faziki. Alzheimer detection based on multi-agent systems : An intelligent image processing environment. In Mostafa Ezziyyani, editor, *Advanced Intelligent Systems for Sustainable Development (AI2SD'2018)*, pages 314–326. Springer International Publishing, 2019. https://doi.org/10.1007/978-3-030-11884-6_28.
- [14] Hanane Alloui, Mohamed Sadgal, and Aziz Elfazziki. Optimized control for medical image segmentation : improved multi-agent systems agreements using particle swarm optimization. *Journal of Ambient Intelligence and Humanized Computing*, 12 :8867–8885, 2021. <https://doi.org/10.1007/s12652-020-02682-9>.
- [15] Youcef Djenouri, Gautam Srivastava, Anis Yazidi, and Jerry Chun-Wei Lin. An edge-driven multi-agent optimization model for infectious disease detection. *Applied Intelligence*, 2022. <https://doi.org/10.1007/s10489-021-03145-0>.
- [16] Hanane Alloui, Mazin Abed Mohammed, Narjes Benameur, Belal Al-Khateeb, Karrar Hameed Abdulkareem, Begonya Garcia-Zapirain, Robertas Damaševičius, and Rytis Maskeliūnas. A multi-agent deep reinforcement learning approach for enhancement of covid-19 ct image segmentation. *Journal of Personalized Medicine*, 12(2), 2022. <https://www.mdpi.com/2075-4426/12/2/309>.
- [17] Carla Pereira, Diana Veiga, Jason Mahdjoub, Zahia Guessoum, Luís Gonçalves, Manuel Ferreira, and João Monteiro. Using a multi-agent system approach for microaneurysm detection in fundus images. *Artificial Intelligence in Medicine*, 60(3) :179–188, 2014. <https://doi.org/10.1016/j.artmed.2013.12.005>.
- [18] Abrar Sundus, Loo Chu, Kiong, and Kubota Naoyuki. A multi-agent approach for personalized hypertension risk prediction. *IEEE Access*, 9 :75090 – 75106, 2021. <https://doi.org/10.1109/ACCESS.2021.3074791>.
- [19] Dr Laurent Leininger. Anatomie de l'œil. <https://dr-leininger.fr/loeil-et-la-vision/anatomie-de-loeil>, 2013. consulte(11/03/2022).
- [20] C. Eswaran Ahmed Wasif Reza. A decision support system for automatic screening of non-proliferative diabetic retinopathy. *Journal of Medical Systems*, 35 :17–24, 2011. <https://doi.org/10.1007/s10916-009-9337-y>.

- [21] Wong Li Yun, U. Rajendra Acharya, Y.V. Venkatesh, Caroline Chee, Lim Choo Min, and E.Y.K. Ng. Identification of different stages of diabetic retinopathy using retinal optical images. *Information Sciences*, 178(1) :106–121, 2008. <https://doi.org/10.1016/j.ins.2007.07.020>.
- [22] Bin Ding, Huimin Qian, and Jun Zhou. Activation functions and their characteristics in deep neural networks. In *2018 Chinese Control And Decision Conference (CCDC)*, pages 1836–1841, 2018.
- [23] What is diabetes. <https://www.idf.org/aboutdiabetes/what-is-diabetes.html>, 2020. consulte(10/03/2022).
- [24] Diabetes complications. <https://www.idf.org/aboutdiabetes/complications.html>, May 2020. consulte(10/03/2022).
- [25] Anatomie de l’oeil. <https://www.oph78.fr/ophtalmologie/anatomie-oeil/#conjonctive>. consulte(11/03/2022).
- [26] Erma Rahayu Mohd Faizal Abdullah Ghulam Mujtaba Rashid Jahangir Hafiz Yasir Ghafoor Uzair Ishtiaq, Sameem Abdul Kareem. Diabetic retinopathy detection through artificial intelligent techniques : a review and open issues. *Multimedia Tools and Applications*, 79 :15209–15252, 2020. <https://doi.org/10.1007/s11042-018-7044-8>.
- [27] FEROU AMEL. *ANALYSE DES IMAGES COULEUR DU FOND D’OEIL POUR L’AIDE AU DIAGNOSTIC EN OPHTALMOLOGIE : APPLICATION A LA DETECTION DES PATHOLOGIES RETINIENNES*. PhD thesis, UNIVERSITE ABOU—BAKR—BELKAID TLEMEN, May 2014. <http://dspace.univ-tlemcen.dz/handle/112/5116>.
- [28] Pavel Hamet and Johanne Tremblay. Artificial intelligence in medicine. *Metabolism*, 69 :S36–S40, 2017. <https://doi.org/10.1016/j.metabol.2017.01.011>.
- [29] John McCarthy. What is artificial intelligence? *Computer Science Department, Stanford University*, 2004. <http://www-formal.stanford.edu/jmc/>.
- [30] What is machine learning? machine learning for beginners. <https://www.edureka.co/blog/what-is-machine-learning/>.
- [31] AD Dongare, RR Kharde, Amit D Kachare, et al. Introduction to artificial neural network. *International Journal of Engineering and Innovative Technology (IJEIT)*, 2(1) :189–194, 2012. <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.1082.1323rep=rep1type=pdf>.
- [32] Daniel Shu Wei Ting, Carol Yim-Lui Cheung, Gilbert Lim, Gavin Siew Wei Tan, Nguyen D Quang, Alfred Gan, Haslina Hamzah, Renata Garcia-Franco, Ian Yew San Yeo, Shu Yen Lee, Edmund Yick Mun Wong, Charumathi Sabanayagam, Mani Baskaran, Farah Ibrahim, Ngiap Chuan Tan, Eric A Finkelstein, Ecosse L Lamoureux, Ian Y Wong, Neil M Bressler, Sobha Sivaprasad, Rohit Varma, Jost B Jonas, Ming Guang He, Ching-Yu Cheng, Gemmy Chui Ming Cheung, Tin Aung, Wynne Hsu, Mong Li Lee, and Tien Yin Wong. Development and validation of a deep learning system for diabetic retinopathy and related eye diseases using retinal images from multiethnic populations with diabetes. *JAMA*, 318(22) :2211–2223, 2017. doi : 10.1001/jama.2017.18152.
- [33] Émilien BARBAUD, Julie QUEIROS, Matthieu PERREIRA, Mona ABID, and Équipe IPI-LS2N. Deep learning sur des données 3d. 2020. https://www.emilienbarbaud.fr/resources/rapport_TER_2020.pdf.

- [34] TOM KELDENICH. Fonction d'activation, comment ça marche? – une explication simple. <https://inside-machinelearning.com/fonction-dactivation-comment-ca-marche-une-explication-simple/>, 2021. consulte(06/09/2022).
- [35] Kaggle dataset [en ligne]. <https://kaggle.com/c/diabetic-retinopathy-detection>.
- [36] Simonyan Karen and Zisserman Andrew. Very deep convolutional networks for large-scale image recognition. *ArXiv*, 1409(1556), 2014. <https://doi.org/10.48550/arXiv.1409.1556>.
- [37] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6) :84–90, 2017. <https://doi.org/10.1145/3065386>.
- [38] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. https://www.cv-foundation.org/openaccess/content_cvpr2016/html/SzegedyRethinking_the_inception_CVPR2016_paper.html.
- [39] Prasanna Porwal, Samiksha Pachade, Ravi Kamble, Manesh Kokare, Girish Deshmukh, Vivek Sahasrabudhe, and Fabrice Meriaudeau. Indian diabetic retinopathy image dataset (idrid) : A database for diabetic retinopathy screening research. *Data*, 3(3), 2018. 10.3390/data3030025.
- [40] Ddr dataset. <https://github.com/nkicsl/DDR-dataset>.
- [41] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. https://www.cv-foundation.org/openaccess/content_cvpr2015/html/SzegedyGoing_Deeper_with_2015_CVPR_paper.html.
- [42] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. https://openaccess.thecvf.com/content_cvpr2017/html/HuangDensely_Connected_Convolutional_CVPR2017_paper.html.
- [43] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. https://openaccess.thecvf.com/content_cvpr2016/html/HeDeep_Residual_Learning_CVPR2016_paper.html.
- [44] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. https://openaccess.thecvf.com/content_cvpr2018/html/HuSqueeze_and_Excitation_Networks_CVPR2018_paper.html.
- [45] Etienne Decencière, Xiwei Zhang, Guy Cazuguel, Bruno Lay, Béatrice Cochener, Caroline Trone, Philippe Gain, Richard Ordonez, Pascale Massin, Ali Erginay, Béatrice Charton, and Jean-Claude Klein. Feedback on a publicly distributed image database : The messidor database. *Image Analysis Stereology*, 33(3), 2014. <https://doi.org/10.5566/ias.1155>.
- [46] T. Aspelund, Ó. Þórisdóttir, E. Ólafsdóttir, A. Gudmundsdóttir, A. B. Einarisdóttir, J. Mehlsen, S. Einarsson, Ó. Pálsson, G Einarsson, T. Bek, and E. Stefánsson. Individual risk assessment

- and information technology to optimise screening frequency for diabetic retinopathy. *Diabetologia*, 54, 2011. <https://doi.org/10.1007/s00125-011-2257-7>.
- [47] Levi Waldron, Melania Pintilie, Ming-Sound Tsao, Frances A Shepherd, Curtis Huttenhower, and Igor Jurisica. Optimized application of penalized regression methods to diverse genomic data. *Bioinformatics (Oxford, England)*, 27(24) :3399–406, 2011. 10.1093/bioinformatics/btr591.
- [48] Ferber J. Les systèmes multi-agents : un aperçu général. *Technique et Science Informatiques*, 16(8) :979–1012, 1997.
- [49] Dr. Toufik MARIR. Les systèmes multi-agents. <http://www.univ-oeb.dz/fsesnv/wp-content/uploads/2019/04/Cours-SMA-Complet.pdf>.
- [50] Jason Mahdjoub. *Vers un système de vision auto-adaptatif à base de systèmes multi-agents*. Theses, Université de Reims - Champagne Ardenne, December 2011. https://tel.archives-ouvertes.fr/tel-00937422/file/these_jason_mahdjoub_pour_cran.pdf.
- [51] Ilemona S.Atawodi. *A Machine Learning Approach to Network Intrusion Detection System Using K Nearest Neighbor and Random Forest*. PhD thesis, Thèse de master : université de Southern Mississippi, 2019. https://aquila.usm.edu/masters_theses/651/.
- [52] Spyder. <https://www.spyder-ide.org/>.
- [53] Raphael Vallat. Pingouin : statistics in python. *Journal of Open Source Software*, 3(31) :1026, 2018. 10.21105/joss.01026.
- [54] Spade documentation. https://spade-mas.readthedocs.io/_/downloads/en/latest/pdf/, 2022. consulte(01/09/2022).
- [55] Matplotlib : Visualization with python. <https://matplotlib.org/>.
- [56] pandas : powerful python data analysis toolkit. <https://pandas.pydata.org/pandas-docs/version/0.7.3/pandas.pdf>, 2012.
- [57] Davis David. The six key things you need to know about scikit-plot. <https://towardsdatascience.com/the-six-key-things-you-need-to-know-about-scikit-plot-119cbd869a> 2020. consulte(11/09/2022).
- [58] Peter Goldsborough. A tour of tensorflow, 2016. 10.48550/ARXIV.1610.01178.
- [59] Navin Kumar Manaswi. *Understanding and Working with Keras*, pages 31–43. Apress, Berkeley, CA, 2018. https://doi.org/10.1007/978-1-4842-3516-4_2.
- [60] Ahlam Rashid. “diabetes dataset”. *Mendeley Data*, Version 1, 2020. doi : 10.17632/wj9rwkp9c2.1.
- [61] Alice Larroumet. Réduction majeure d’hba1c et rétinopathie dans le diabète de type 2. *Sciences du Vivant [q-bio]*, 2020. dumas-03008066.

Résumé

La Rétinopathie Diabétique (RD) est une complication redoutée du diabète qui peut causer la cécité ou autrement dit la perte de vue. Dans le domaine de la santé, le traitement des maladies est plus efficace lorsqu'il est détecté à un stade précoce d'où l'importance et l'efficacité du diagnostic pour la détection mais aussi la prévention de cette pathologie. Pour les patients atteints de diabète, il existe plusieurs facteurs favorisant le développement de la RD. A cet effet, l'exploitation et l'analyse de ces facteurs au moment opportun nous permet de prédire le risque d'apparition de la rétinopathie diabétique.

Pour ce faire, les techniques de l'apprentissage automatique, très utilisées dans le domaine de la santé, représentent une des meilleures alternatives pour développer un bon modèle de prédiction. Cependant, ces méthodes s'avèrent insuffisantes dans un environnement dynamique et complexe. L'IA notamment les systèmes multi-agents ont révélé leur capacité à prendre en charge ce type de systèmes.

Dans ce travail, nous proposons de combiner l'apprentissage automatique et les systèmes multi-agents pour mettre au point un modèle qui permet de prédire le risque d'apparition de la rétinopathie diabétique chez un sujet diabétique. Ce système est composé de trois agents communicants dont l'architecture interne de deux d'entre eux est un réseau de neurones. Les résultats d'expérimentations obtenus sont très prometteurs.

Abstract

Diabetic Retinopathy (DR) is a serious complication of diabetes that can cause blindness or in other words, loss of sight. It is effective when detected at an early stage, hence the importance and effectiveness of diagnosis for the detection but also the prevention of this pathology. For patients with diabetes, there are several factors favoring the development of DR. To this end, the exploitation and analysis of these factors at the appropriate time allows us to predict the risk of the appearance of diabetic retinopathy.

For this purpose, machine learning techniques, widely used in the health field, represent one of the best alternatives for developing a good prediction model. However, these methods are insufficient in a dynamic and complex environment. AI, especially multi-agent systems, have revealed their capacity to handle this type of systems.

In this work, we propose to combine machine learning and multi-agent systems to develop a model that can predict the risk of occurrence of diabetic retinopathy in a diabetic patient. This system is composed of three agents communicating with each other whose internal architecture is a neural network. The obtained experimental results of the proposed model are very promising.

