

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
Université Abderrahmane Mira de Béjaïa



Université A. Mira de Béjaïa  
Faculté des Sciences Exactes  
Département d'Informatique

## Mémoire de Fin de Cycle

En vue de l'obtention du diplôme de MASTER recherche en Informatique  
Option : Intelligence Artificielle

### Thème

Renforcement de la sécurité routière dans les  
systèmes de transport intelligents

*Présenté par :*

M<sup>lle</sup> MEDKOUR Amel  
M<sup>lle</sup> BENALAOUA Ibtissam

*Soutenu devant le jury composé de :*

<i>Présidente</i>	Mme. CHABANE Sarah	M.C.B	U. de Béjaïa
<i>Examineur</i>	M. AMROUN Kamal	Professeur	U. de Béjaïa
<i>Encadrant</i>	M. AISSANI Sofiane	M.C.A	U. de Béjaïa
<i>Co-Encadrante</i>	Mme. ZIZI Kahina	Doctorante	U. de Béjaïa
<i>Invitée</i>	M <sup>lle</sup> . CHENACHE Tinhinane	Doctorante	U. de Béjaïa

Promotion 2023 – 2024

## *Remerciements*

*Au terme de ce travail, nous tenons à exprimer notre profonde gratitude et nos sincères remerciements.*

Nous remercions Dieu le Tout-Puissant de nous avoir donné la force, la volonté de donner le meilleur de nous-mêmes et le courage de mener ce travail à bien.

Nous tenons en premier lieu à exprimer notre profonde reconnaissance à notre encadrant, **M. AISSANI Sofiane**, pour son encadrement, sa confiance, son accompagnement, son encouragement et ses conseils tout au long de notre cursus.

Nous tenons également à remercier très chaleureusement notre co-encadrante, **Mme ZIZI Kahina**, pour sa collaboration précieuse pendant la rédaction de ce mémoire, ses encouragements et ses conseils avisés.

Nous adressons nos remerciements sincères aux membres du jury pour avoir consacré leur temps à la lecture et à la correction de ce mémoire.

Nos remerciements les plus vifs vont tout particulièrement à nos parents, qui ont été une source inépuisable de courage, de volonté et de confiance. Nous leur serons éternellement reconnaissants.

Enfin, nous exprimons toute notre gratitude à tous les membres du département d'Informatique de l'Université de Béjaia, qu'ils soient enseignants ou cadres administratifs, qui ont de près ou de loin, contribué à ce que notre formation et nos travaux se déroulent dans les meilleures conditions.

*Amel & Ibtissam*

## *Dédicaces*

Merci à Allah de m'avoir donné la capacité d'écrire et de réfléchir,  
la force d'y croire, et la patience d'aller jusqu'au bout du rêve.

À mes chers parents Khoudir et Fatima, si dévoués,  
Pour vos sacrifices, votre amour sacré,  
Votre tendresse et votre soutien sans fin,  
Vos prières qui m'ont guidé chaque matin.

À mes frères Islam, Amine, Ghilas,  
J'espère être un exemple pour vous, comme il se doit.

À mes sœurs Katia, Taoues, Zineb, proches de cœur,  
Votre soutien et votre amour, une source de bonheur.

À mes amis fidèles, présents dans chaque épreuve,  
Votre amitié précieuse, un trésor qui s'élève.

À vous tous qui avez illuminé mon chemin,  
Ce mémoire vous est dédié, tel un doux refrain.  
Que chaque mot, chaque ligne, chaque idée exprimée,  
Porte l'écho de notre lien, de nos moments partagés.  
Merci d'être là, de m'inspirer chaque jour,  
Votre présence est un cadeau, un amour sans détour.

À toute ma famille,  
À tous ceux que j'aime et qui m'aiment sans exception,  
À moi-même.

## *Dédicaces*

*Tout d'abord, je tiens à remercier DIEU de m'avoir donné la force et le courage de mener à bien ce modeste travail.*

***Je tiens à dédier cet humble travail à :***

*À l'âme de mon grand-père B. Said, décédé le 08/01/2015. Ta perte a créé un vide immense dans ma vie.*

*À mon père Abdel Moumene et ma mère IDIREN Nadia, qui ont toujours été ma source d'inspiration, de soutien et d'amour inconditionnel. Votre confiance en moi a été le moteur de ma persévérance.*

*À ma soeur Tassedra et mon frère Ghilas pour leurs encouragements.*

*À toute ma famille. Aucun langage ne saurait exprimer mon respect et ma considération pour votre soutien et vos encouragements.*

*À mes copines Macilia, Lyna, Chaima, Tinhinane, Bouchra, Cylia, Lyly, Lydia et Nadia. Je ne peux pas trouver les mots justes et sincères pour vous exprimer mon affection. Vous êtes pour moi des soeurs et des amies sur qui je peux compter.*

*À mes amis Zaki, Nabil, Sofiane et Omar.*

*Pour finir, à tous ceux qui m'aiment et que j'aime.*

# Table des matières

<b>Liste des figures</b>	<b>iii</b>
<b>Liste des tableaux</b>	<b>iv</b>
<b>Liste des abréviations</b>	<b>iv</b>
<b>1 Notions fondamentales</b>	<b>3</b>
1.1 Introduction	3
1.2 Système de Transport Intelligent (STI)	3
1.2.1 Définition des STI	3
1.2.2 Technologies des STI	4
1.2.3 Objectifs et Avantages des STI	5
1.2.4 Défis des STI	5
1.3 Intelligence artificielle (IA)	6
1.3.1 Apprentissage automatique	7
1.4 Conclusion	13
<b>2 Etat de l'art sur les comportements dangereux de conduite</b>	<b>14</b>
2.1 Introduction	14
2.2 Classification des études précédentes	14
2.3 Étude des travaux existants sur la détection et la classification des comportements de conduite dangereux dans les STIs	15
2.3.1 Solution basée sur l'étude des comportements du conducteur	15
2.3.2 Solution basée sur l'étude des comportements du véhicule	18
2.3.3 Solution basée sur l'étude des comportements hybrides	24
2.4 Comparaison des travaux étudiés	26
2.4.1 Critère de comparaison	26
2.4.2 Tableau comparatif	27
2.4.3 Discussion	31
2.5 Conclusion	32
<b>3 Approche proposée (LSTM-RF)</b>	<b>33</b>
3.1 Introduction	33
3.2 Motivation	33
3.3 Notre Proposition	34

3.4	UAH-DriveSet . . . . .	34
3.5	Phase de préparation des données . . . . .	35
3.5.1	Prétraitement des données . . . . .	36
3.6	Phase de détection avec LSTM . . . . .	42
3.7	Phase de classification avec RF . . . . .	44
3.8	Conclusion . . . . .	45
<b>4</b>	<b>Expérimentation</b>	<b>46</b>
4.1	Introduction . . . . .	46
4.2	Outils de développement . . . . .	46
4.3	Bibliothèques utilisées . . . . .	47
4.4	Évaluation des performances du modèle . . . . .	47
4.5	Evaluation des performances . . . . .	49
4.5.1	Phase de détection . . . . .	49
4.5.2	Phase de classification . . . . .	52
4.6	Conclusion . . . . .	55
	<b>Conclusion générale et perspectives</b>	<b>56</b>
	<b>Bibliographie</b>	<b>57</b>

# Table des figures

1.1	Système de transport intelligent ITS [44]. . . . .	4
1.2	Schéma sur les différents domaines de l'IA. . . . .	6
1.3	Exemple de processus d'Apprentissage Automatique (reconnaissance d'objets) [32]. . . . .	7
1.4	Exemple de processus d'apprentissage profond (reconnaissance d'objets)[32]. . . . .	7
1.5	Structure de modèle de réseau de neurones artificiels. . . . .	8
1.6	Architecte d'un perceptron multicouche. . . . .	9
1.7	Architecture d'un réseau de neurone convolutif [5]. . . . .	9
1.8	Structure d'un réseaux de neurones récurrents [40]. . . . .	10
1.9	La structure d'un LSTM [19]. . . . .	10
1.10	Structure d'un arbre de décision. . . . .	11
1.11	Structure d'une forêt aléatoire avec combinaison directe des prédictions. . . . .	12
2.1	Classification des travaux passés en revue. . . . .	15
3.1	Structure de la solution proposée. . . . .	34
3.2	Attributs du dataset. . . . .	36
3.3	Illustration de la distance de sécurité entre les véhicules. . . . .	38
3.4	Illustration de la détection des zigzags. . . . .	40
3.5	Attributs du nouveau dataset. . . . .	41
3.6	Diagramme de détection par LSTM. . . . .	43
3.7	Diagramme de classification par RF. . . . .	45
4.1	Taux d'erreur et précision du LSTM. . . . .	50
4.2	Taux d'erreur et précision du CNN. . . . .	51
4.3	Taux d'erreur et précision MLP. . . . .	51
4.4	La matrice de confusion pour le modèle LSTM-RF. . . . .	53
4.5	La courbe ROC pour la classe 0 (normal). . . . .	54
4.6	La courbe ROC pour la classe 1 (anormal). . . . .	54

# Liste des tableaux

2.1	Tableau comparatif. . . . .	30
3.1	Attributs du UAH-DriveSet. . . . .	35
4.1	Performance des modèles selon les métriques d'évaluation . . . . .	52
4.2	Performance des modèles selon les métriques d'évaluation . . . . .	55

# Liste des abréviations

<b>STI</b>	Système de Transport Intelligent
<b>IA</b>	Intelligence Artificielle
<b>ML</b>	Machine Learning
<b>DL</b>	Deep Learning
<b>RNA</b>	Réseaux de Neurones Artificiels
<b>MLP</b>	Multi Layer Perceptron
<b>CNN</b>	Convolutional Neural Network
<b>RNN</b>	Recurrent Neural Network
<b>LSTM</b>	Long Short-Term Memory
<b>AD</b>	Arbre de décision
<b>RF</b>	Random Forest
<b>RL</b>	Régression Logistique
<b>SAX</b>	Symbolic Aggregate approxImation
<b>SAX-DF</b>	Symbolic Aggregate approxImation based Data Fusion mode
<b>KNN</b>	Convolutional Neural Network
<b>ML</b>	K-Nearest Neighbors
<b>SVM</b>	Support Vector Machine
<b>MSE</b>	Mean Square Error
<b>MAE</b>	Mean Absolute Error
<b>RMSE</b>	Root Mean Squared Error

# Introduction générale

Les systèmes de transport intelligents (STI) ont connu une expansion spectaculaire au cours des dernières décennies, révolutionnant la façon dont nous concevons, gérons et vivons la mobilité urbaine. Ces systèmes intègrent une multitude de technologies avancées telles que les capteurs, les communications sans fil, et l'intelligence artificielle pour optimiser la sécurité, l'efficacité et la durabilité des déplacements. Malgré ces progrès, la sécurité routière demeure une préoccupation majeure, avec des comportements de conduite inappropriés représentant une menace constante pour les usagers de la route et la gestion du trafic.

Les accidents de la route représentent un problème qui affecte gravement la société actuelle. Chaque année, des millions de vies sont bouleversées par des collisions entraînant des décès, des blessures graves et des dommages matériels importants. Cette situation cause non seulement une souffrance humaine immense mais aussi des coûts économiques élevés, incluant les frais médicaux, les pertes de productivité et les dépenses liées à la réparation des infrastructures endommagées [20]. Les causes des accidents sont diverses : excès de vitesse, conduite en état d'ivresse, distractions au volant, non-respect des règles de circulation, et parfois des défaillances techniques des véhicules. Bien que des progrès aient été réalisés en matière de sécurité routière grâce à l'amélioration des technologies de véhicules et à des campagnes de sensibilisation, le nombre d'accidents reste alarmant. Il est donc essentiel de continuer à investir dans des infrastructures sécurisées, à renforcer les réglementations et à promouvoir des comportements responsables pour réduire efficacement le nombre d'accidents.

Pour répondre à cette problématique, notre approche repose sur l'utilisation de techniques avancées d'apprentissage automatique. Nous avons développé un modèle intégrant les capacités des réseaux de neurones récurrents Long Short-Term Memory (LSTM) et des forêts aléatoires (Random Forest), afin de détecter de manière précise et fiable les comportements de conduite à risque. Cette combinaison permet une analyse approfondie des données de conduite en temps réel, facilitant ainsi une intervention proactive pour réduire les accidents et améliorer la sécurité des usagers de la route.

Notre mémoire est organisé en quatre chapitres principaux :

Le premier chapitre est consacré aux Systèmes de Transport Intelligents (STI). Il offre une vue d'ensemble des STI, définissant leurs technologies sous-jacentes, leurs objectifs et avantages, ainsi que les défis qu'ils rencontrent. Nous explorons également le rôle de l'intelligence artificielle dans les STI, en particulier les techniques d'apprentissage automatique.

Le deuxième chapitre approfondit dans les comportements de conduite leur compréhension et leur détection. En examinant les différentes méthodes proposées dans la littérature, ce chapitre explore les approches et les techniques les plus récentes et efficaces pour détecter et classer les comportements de conduite dangereux. Cette analyse critique permet d'identifier les forces et les limitations des méthodes existantes, fournissant ainsi une base solide pour le développement de notre propre modèle de détection et classification des comportements de conduite.

Dans le troisième chapitre, nous développerons la conception de notre système de prédiction qui combine les réseaux de neurones récurrents Long Short-Term Memory (LSTM) et les forêts aléatoires (Random Forest). Nous explorerons en détail les différentes étapes de notre approche, depuis la préparation des données jusqu'à l'architecture et l'entraînement des modèles. Ce chapitre vise à approfondir la

mise en oeuvre de notre méthodologie afin de fournir une prédiction précise et fiable des comportements de conduite dangereux.

Le quatrième chapitre est dédié à l'expérimentation et à l'évaluation de notre modèle de prédiction des comportements de conduite. Nous mènerons une analyse approfondie en comparant les résultats obtenus par notre modèle avec ceux d'autres approches existantes. Cette évaluation comparative nous permettra d'apprécier l'efficacité de notre approche et de démontrer sa valeur dans le domaine de la sécurité routière et des systèmes de transport intelligents.

Ce travail sera terminé par une conclusion générale qui permettra d'examiner nos idées et nos résultats en donnant des améliorations possibles et quelques perspectives.

# 1

## Notions fondamentales

### 1.1 Introduction

Les systèmes de transport intelligents (STI) intègrent des technologies de sécurité routière pour réduire les accidents et améliorer la sécurité des usagers de la route. Ce chapitre explore les principes fondamentaux des STI, leur définition et leur objectif principal : améliorer la sécurité et optimiser la gestion du transport. Nous examinerons également les technologies clés qui sous-tendent les STI et leur rôle essentiel dans cette quête. Nous aborderons ensuite les objectifs, les avantages et les défis des STI dans le contexte du transport moderne. En parallèle, nous plongerons dans le domaine de l'intelligence artificielle (IA) et son implication croissante dans les STI. Nous explorerons différentes branches de l'IA, notamment l'apprentissage automatique, en mettant un accent particulier sur certaines approches de l'apprentissage profond.

### 1.2 Système de Transport Intelligent (STI)

#### 1.2.1 Définition des STI

Le Système de Transport Intelligent (STI) est un système global de gestion des services de transport. Il est qualifié d'intelligent car il tire parti de capacités liées à l'intelligence, telles que le traitement de l'information. Ainsi, il permet le traitement, l'analyse et la communication des données relatives à ces services. Les technologies envisagées couvrent un large spectre, allant de la télématique dans le domaine du transport, à l'utilisation de l'électronique, qu'elle soit embarquée ou fixe (tels que les capteurs et les moyens de calcul), en passant par les télécommunications, les bases de données et d'informations, les systèmes de régulation, ainsi que les paiements électroniques [31].

Ces applications englobent tous les moyens de transport, qu'ils soient routiers, ferroviaires, aériens ou maritimes. Leur objectif est de garantir la sécurité et de réguler les flux de circulation, tout en fournissant des informations aux utilisateurs des transports en commun et du transport de marchandises [31].

Notre étude se focalise spécifiquement sur les systèmes de transport routier(voir la figure 1.1).

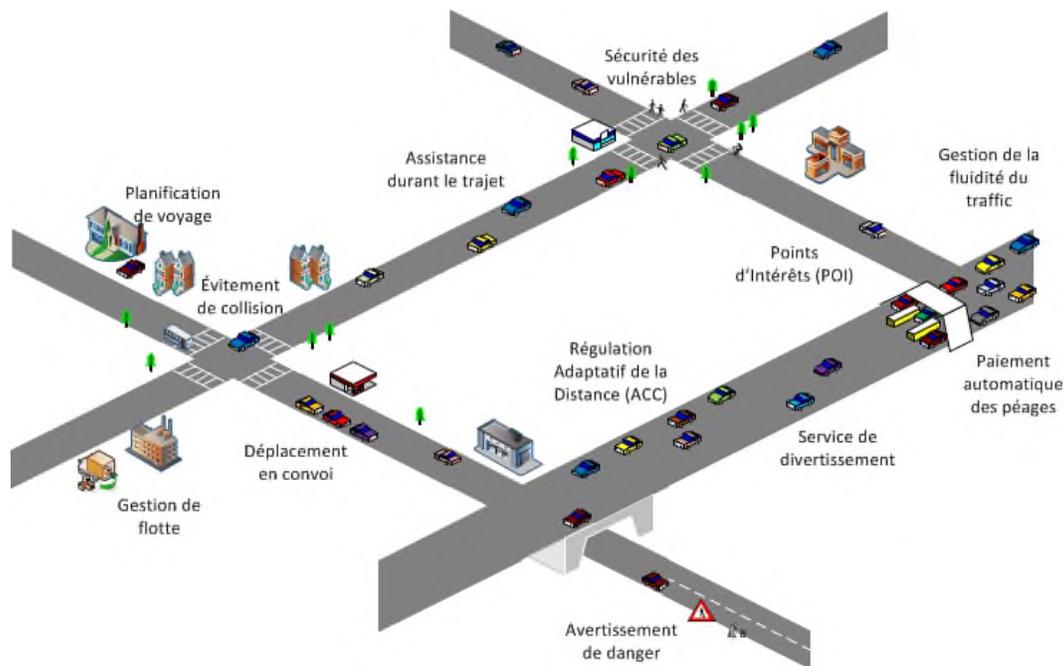


FIGURE 1.1 – Système de transport intelligent ITS [44].

## 1.2.2 Technologies des STI

Les systèmes de transport intelligents (STI) font appel à une gamme variée de technologies, allant des systèmes de gestion de base tels que les contrôles des feux de signalisation, la gestion des conteneurs, les panneaux à messages variables, les radars automatiques et la vidéo surveillance, aux applications plus avancées intégrant des données en temps réel en provenance de multiples sources, telles que les informations météorologiques, les systèmes de dégivrage des ponts et les systèmes de navigation embarqués fournissant des données de temps de parcours en temps réel. De plus, des techniques prédictives sont développées pour permettre une modélisation avancée et une comparaison avec une base de données historiques de référence [7].

Quelques-unes des technologies typiquement intégrées dans les STI sont décrites dans les sections qui suivent :

### 1.2.2.1 Technologies de localisation

Les technologies de localisation se basent sur des systèmes de positionnement, avec deux catégories principales :

- Géolocalisation par système de positionnement par satellites comme GPS et Galileo.
- Téléphonie mobile [7].

### 1.2.2.2 Technologies de capteurs

Les avancées technologiques dans le domaine des capteurs ont facilité le développement de systèmes de transport intelligents grâce à des données fiables, abondantes et fréquentes. Ces technologies permettent de mesurer les caractéristiques des véhicules ainsi que le flux de circulation. Les capteurs les plus couramment utilisés incluent les capteurs vidéo, les boucles électromagnétiques et les réseaux de capteurs sans fil [7].

### 1.2.2.3 Technologies de calcul avancées

Les technologies de calcul avancées dans les systèmes de transport intelligents englobent l'utilisation de processeurs informatiques plus performants et de plates-formes embarquées modernes. Ces avancées permettent l'intégration d'applications logicielles sophistiquées, notamment basées sur la modélisation, l'intelligence artificielle et l'informatique omniprésente. Elles jouent un rôle essentiel dans l'amélioration des fonctionnalités des systèmes de transport intelligents, en rendant les véhicules plus efficaces et sûrs [7].

## 1.2.3 Objectifs et Avantages des STI

Les systèmes de transport intelligents (STI) sont déployés dans le but d'atteindre plusieurs objectifs tout en offrant divers avantages pour améliorer la gestion et la performance des réseaux de transport. Ces objectifs incluent notamment :

- Amélioration de la sécurité routière et de la fluidité du trafic : Les STI intègrent des technologies avancées, comme les systèmes d'aide à la conduite et les alertes de collision, pour réduire les accidents. En fournissant des informations en temps réel sur le trafic, ils aident à réduire la congestion et à optimiser les itinéraires, diminuant ainsi les temps de trajet [53].
- Réduction des émissions polluantes et promotion de modes de transport écologiques : Les STI favorisent l'utilisation de modes de transport plus respectueux de l'environnement, ce qui contribue à réduire les émissions de gaz à effet de serre et la pollution atmosphérique. En optimisant l'efficacité énergétique des déplacements, ils participent également à la diminution de la consommation d'énergie et des émissions de CO<sub>2</sub> [28].
- Amélioration de la qualité de vie des usagers : En rendant les services de transport plus fiables et plus sûrs, les STI réduisent les temps d'attente et le stress des usagers. Cela améliore la qualité de vie en offrant une expérience de voyage plus agréable et en facilitant l'accès aux différents lieux.

## 1.2.4 Défis des STI

L'économie mondiale dépend de réseaux de transport efficaces, essentiels pour le déplacement sécurisé et ponctuel des passagers et des marchandises. Cependant, les réseaux de transport routier actuels peinent souvent à atteindre ces objectifs. Les principaux défis du transport moderne sont :

- Accidents et sécurité : La hausse du trafic routier entraîne environ 1,35 million de décès et entre 20 et 50 millions de blessures par an dans le monde. Les principales causes sont les comportements de conduite négligents, tels que le non-respect des règles de circulation, la vitesse excessive, la fatigue et la conduite sous l'influence de substances [23].

- Congestion : fléau des zones urbaines, la congestion routière résulte d'un déséquilibre entre la croissance rapide de la propriété de véhicules et le développement insuffisant des infrastructures. Des systèmes de signalisation routière inadéquats, des réseaux routiers mal conçus et des événements imprévus tels que des accidents exacerbent encore ce problème [23].
- Impact environnemental : le transport routier est une source de pollution atmosphérique et sonore, notamment dans les villes à forte densité de véhicules. La combustion inefficace pendant les périodes d'arrêt dans les embouteillages aggrave la qualité de l'environnement.

### 1.3 Intelligence artificielle (IA)

Bien que l'IA soit considérée comme le plus récent domaine révolutionnaire en évolution technologique, le terme a été inventé en août 1956 par "John McCarthy" [29]. Il s'agit d'un processus de simulation de l'intelligence humaine basé sur l'application d'algorithmes mis en oeuvre dans un environnement informatique dynamique pour permettre aux ordinateurs de penser et de se comporter comme des humains [1].

L'IA englobe également les sous-domaines de l'apprentissage automatique et de l'apprentissage profond, qui utilisent des algorithmes pour créer des systèmes experts capables de faire des prédictions ou des classifications basées sur des données entrantes. Comme le montre le schéma présenté à la figure 1.2.

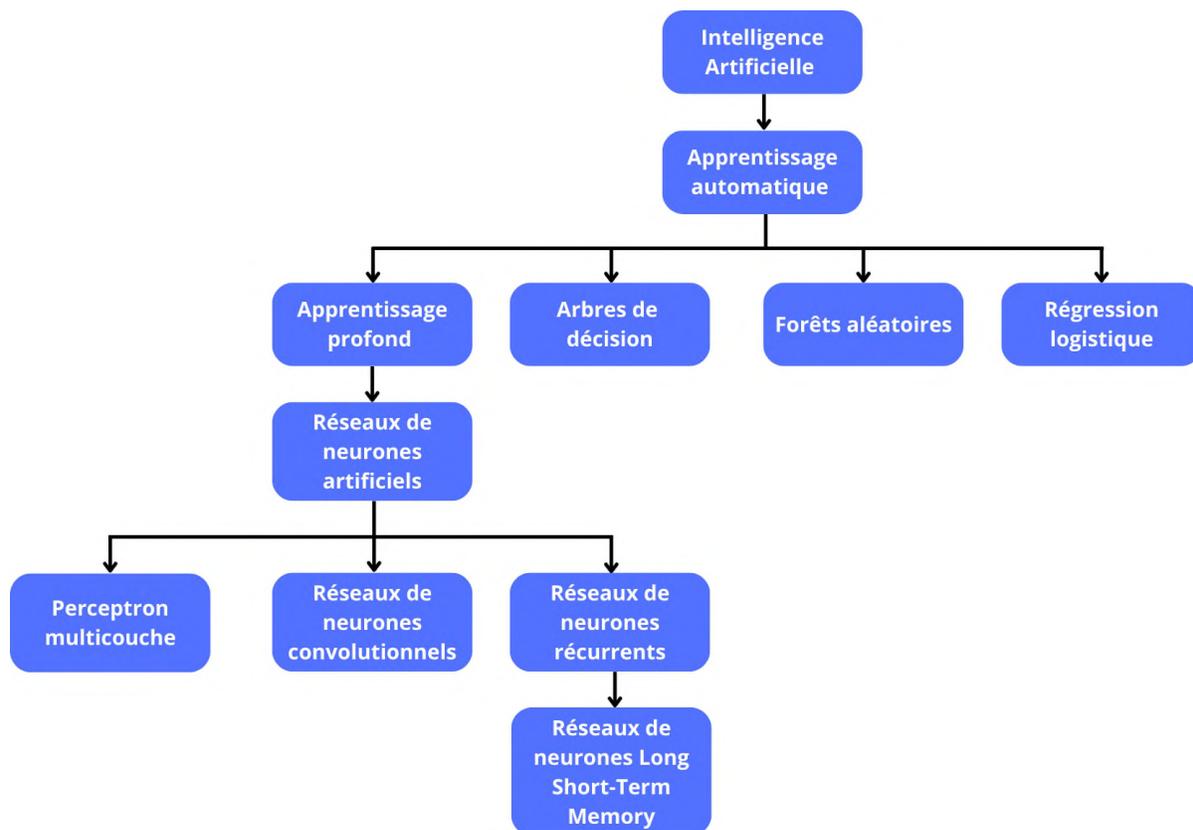


FIGURE 1.2 – Schéma sur les différents domaines de l'IA.

### 1.3.1 Apprentissage automatique

L'apprentissage automatique, ou machine learning, est un domaine de l'intelligence artificielle où les machines apprennent à partir de données sans être explicitement programmées. Ce processus est similaire à l'apprentissage humain, où le cerveau assimile de nouvelles informations. Les réseaux de neurones artificiels, inspirés par le fonctionnement du cerveau (voir figure 1.3), permettent à ces machines d'apprendre en modifiant leurs opérations mathématiques [50].

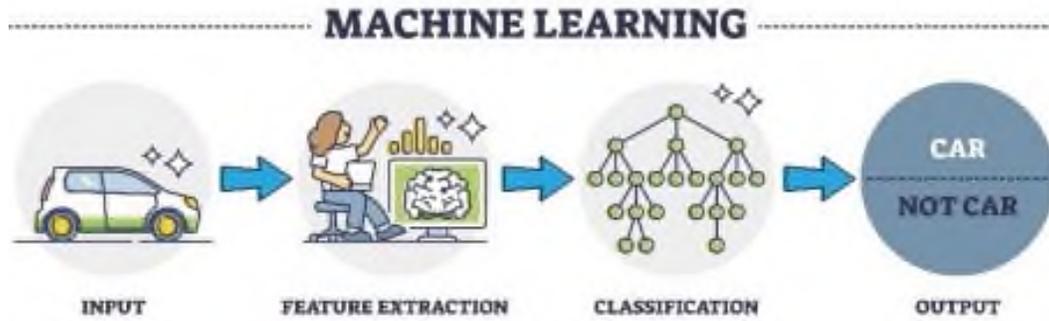


FIGURE 1.3 – Exemple de processus d'Apprentissage Automatique (reconnaissance d'objets) [32].

#### 1.3.1.1 Apprentissage profond

L'apprentissage profond, ou deep learning, est une branche de l'intelligence artificielle qui utilise des réseaux de neurones artificiels pour analyser des données. Ces réseaux, constitués de nombreuses unités appelées neurones, traitent les données en plusieurs couches successives pour extraire des caractéristiques de plus en plus complexes. Par exemple, dans la reconnaissance visuelle, les premières couches identifient des éléments simples comme des lignes et des courbes, tandis que les couches supérieures reconnaissent des formes et des objets (voir la figure 1.4). Les avancées récentes du l'apprentissage profond ont été rendues possibles par l'augmentation de la puissance informatique et l'existence de vastes ensembles de données. Ses applications couvrent divers domaines tels que la reconnaissance d'images, la traduction automatique et la génération de texte [6].

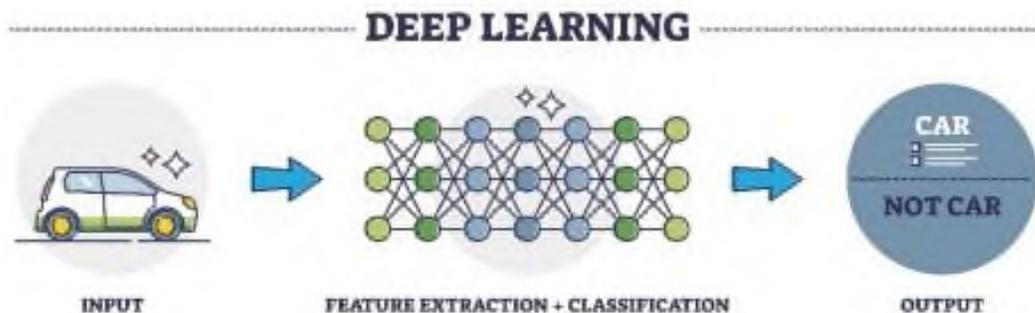


FIGURE 1.4 – Exemple de processus d'apprentissage profond (reconnaissance d'objets)[32].

## A. Réseaux de neurones artificiels

La conception des réseaux de neurones artificiels (RNA) s'inspire de la structure des neurones biologiques du cerveau humain, dans le but de modéliser la manière dont le cerveau traite l'information. Les RNA sont composés d'au moins deux couches de neurones : une couche d'entrée et une couche de sortie, avec généralement une ou plusieurs couches intermédiaires appelées « couches cachées » [9] comme illustré dans la figure 1.5.

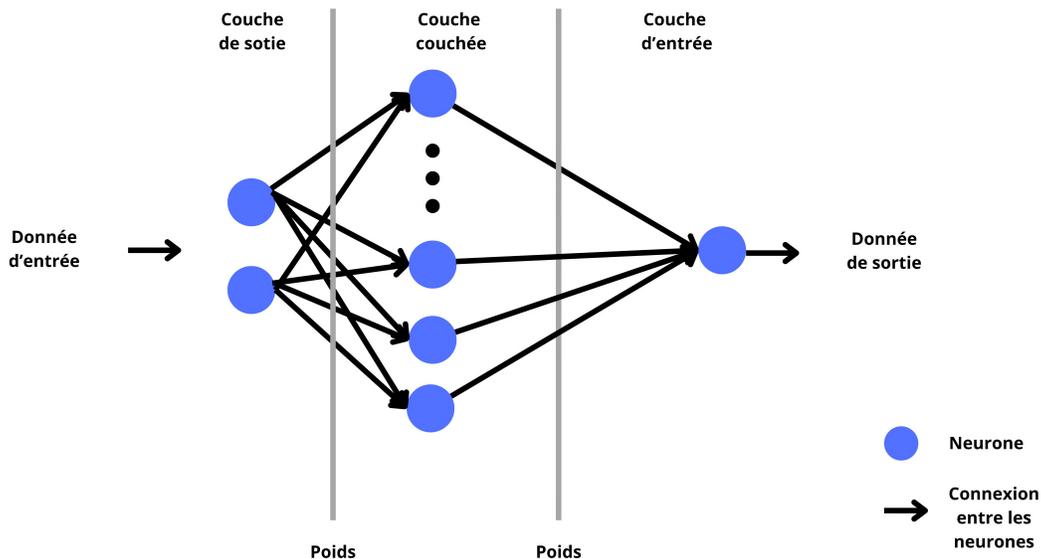


FIGURE 1.5 – Structure de modèle de réseau de neurones artificiels.

Le traitement de l'information dans un RNA suit une séquence définie : les informations sont transmises sous forme de signaux aux neurones de la couche d'entrée, où elles sont traitées. Chaque neurone se voit attribuer un « poids » qui détermine l'importance des informations reçues. Ce poids, associé à une fonction de transfert, détermine quelles informations passent à travers le neurone. Ensuite, une fonction d'activation, souvent la sigmoïde, la tanh ou la ReLU, associée à une valeur seuil, calcule et pondère la valeur de sortie du neurone. Selon cette valeur, différents neurones sont activés et connectés [9].

Ces connexions et pondérations forment un algorithme qui associe un résultat spécifique à chaque entrée. Chaque itération de l'apprentissage permet d'ajuster les poids, améliorant ainsi progressivement la précision et la fiabilité du réseau. Pour garantir que les connexions au sein d'un RNA soient correctement établies, il est nécessaire de procéder à un « entraînement » avec des données d'apprentissage, souvent en utilisant l'algorithme de rétropropagation pour minimiser l'erreur [9].

- **Perceptron multicouche**

Le perceptron multicouche (multi layer perceptron MLP) est un classifieur linéaire de type réseau neuronal formel organisé en plusieurs couches (figure 1.6) au sein desquelles une information circule de la couche d'entrée vers la couche de sortie uniquement. Chaque couche est constituée d'un nombre variable de neurones, les neurones de la couche de sortie correspondant toujours aux sorties du système [6].

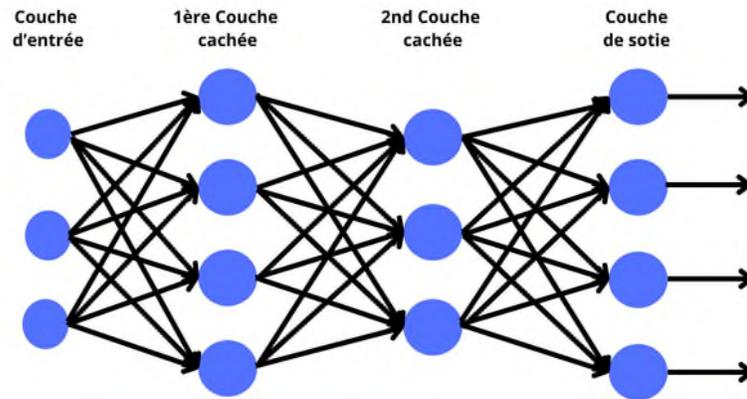


FIGURE 1.6 – Architecture d'un perceptron multicouche.

- **Réseaux de neurones convolutionnels**

Les réseaux neuronaux convolutionnels (CNN) sont un type de réseau neuronal artificiel largement utilisé dans les domaines de l'analyse d'images et de vidéos et de la reconnaissance d'objets. Ce type de réseau neuronal est conçu pour traiter des données qui sont généralement bidimensionnelles et utilise de nombreuses couches différentes pour extraire différents profils d'images et déterminer et classer leur contenu (voir la figure 1.7) [5].

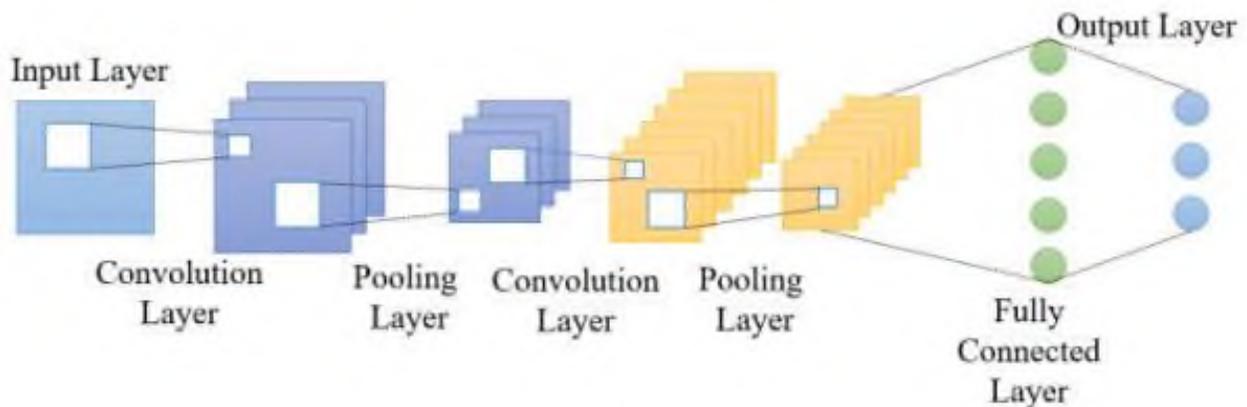


FIGURE 1.7 – Architecture d'un réseau de neurone convolutif [5].

- **Réseaux de neurones récurrents**

Les réseaux de neurones récurrents (RNN) sont une catégorie de réseaux de neurones spécialement conçus pour traiter des séquences d'entrées élément par élément. Ils maintiennent, dans leurs unités cachées (les neurones des couches cachées), un vecteur d'état appelé état caché, qui contient des informations sur l'historique des éléments passés de la séquence. Chaque sortie des unités cachées  $h(t)$  dépend de l'état caché  $h(t) - 1$  comme le montre la figure 1.8. Cet état caché peut être considéré comme une mémoire. En effet, ajouter de la mémoire à un réseau neuronal permet de traiter les informations de la séquence elle-même : l'information séquentielle est préservée dans l'état caché, ce qui permet de trouver des corrélations entre des événements séparés par plusieurs pas de temps. Cette mémoire est contenue dans les couches cachées qui ont une boucle de rétroaction, et par conséquent elles constituent des couches récurrentes [34].

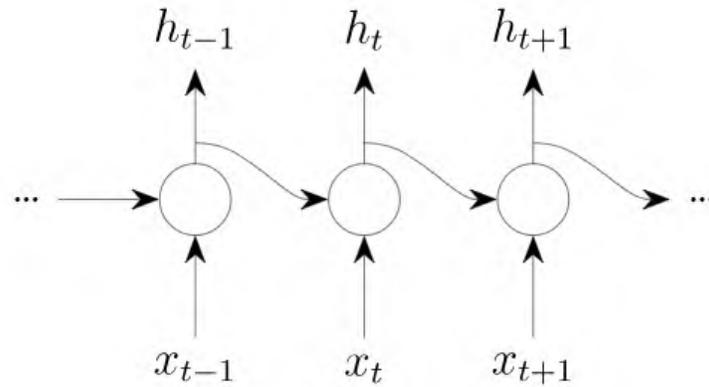


FIGURE 1.8 – Structure d’un réseaux de neurones récurrents [40].

— Réseaux de neurones Long Short-Term Memory

Les Long Short-Term Memory (LSTM) sont un type spécial de réseaux de neurones récurrents (RNN). Ils partagent la même structure en chaîne que les RNN classiques, c’est-à-dire qu’ils traitent les séquences de données de manière itérative, chaque unité (ou cellule) prenant en compte l’information de l’unité précédente ainsi que l’entrée actuelle. Ce qui différencie les LSTM est la structure interne de leur module répétitif, qui permet de mieux gérer les dépendances à long terme [42].

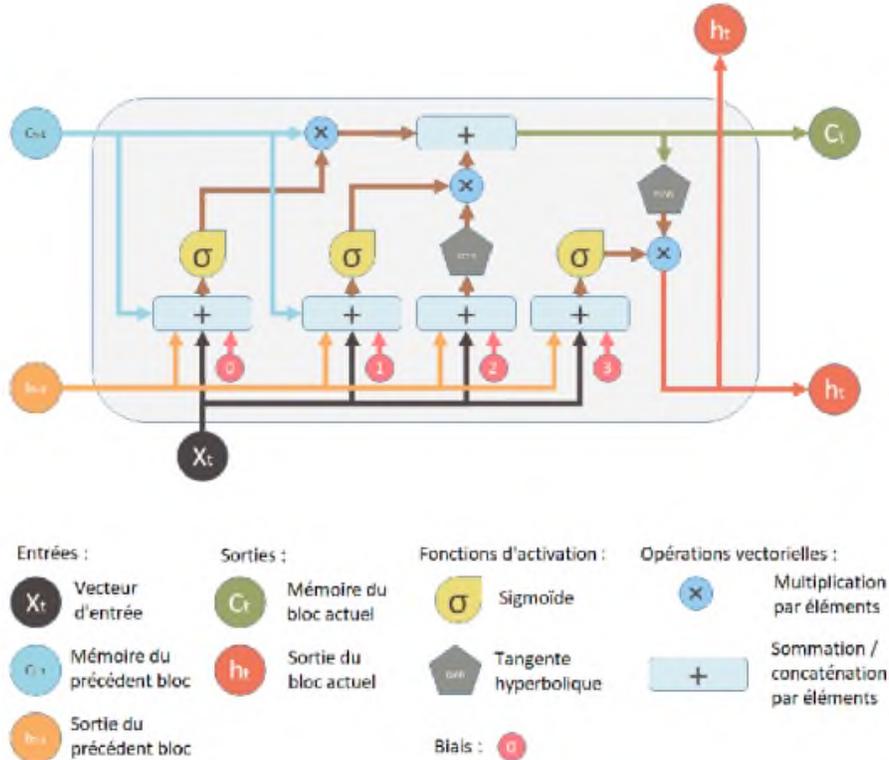


FIGURE 1.9 – La structure d’un LSTM [19].

Les LSTM utilisent trois types de portes pour contrôler le flux d'informations à chaque pas de temps la structure est illustrée par la figure 1.9 :

- Porte d'entrée : Détermine quelles nouvelles informations sont ajoutées à la cellule de mémoire.
- Porte d'oubli : Décide quelles informations de la cellule de mémoire doivent être oubliées.
- Porte de sortie : Choisit quelles informations de la cellule de mémoire sont utilisées pour calculer l'état caché et la sortie à l'étape actuelle [19].

Ces portes sont essentielles pour réguler le flux d'information et permettre au réseau de conserver des informations pertinentes sur de longues périodes de temps, ce qui résout en partie le problème du gradient qui disparaît souvent rencontré dans les RNN classiques [8].

### 1.3.1.2 Arbres de décision

Les arbres de décision sont une technique populaire en machine learning utilisée principalement pour les tâches de classification et de régression. Leur succès est principalement dû à leur capacité à traiter des problèmes complexes et à fournir des solutions interprétables. Ils offrent une représentation facile à comprendre et à interpréter, permettant de dériver des règles de décision logiques [3].

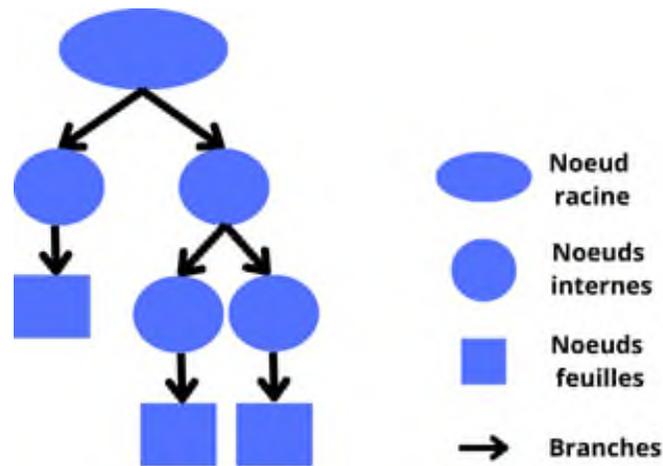


FIGURE 1.10 – Structure d'un arbre de décision.

Un arbre de décision est composé de :

- Noeud racine (Root Node) : Le point de départ de l'arbre. Il n'a pas de branches entrantes et représente la caractéristique (ou attribut) qui divise le jeu de données initial en sous-ensembles plus petits.
- Noeuds internes (Internal Nodes) : Ces noeuds sont situés entre le noeud racine et les noeuds feuille. Ils effectuent des évaluations sur les attributs et dirigent le flux de l'arbre vers les branches correspondantes.
- Branches (Edges) : Les connexions entre les noeuds. Chaque branche représente une valeur spécifique d'un attribut.

- Noeuds feuilles (Leaf Nodes) : Les nœuds terminaux de l'arbre. Ils contiennent les résultats finaux (par exemple, les étiquettes de classe pour une tâche de classification ou les valeurs de sortie pour une régression). Une représentation visuelle de cet arbre, montrant clairement les nœuds racine, internes et feuilles, cela est illustré dans la figure 1.10.

### 1.3.1.3 Forêts aléatoires

Les forêts aléatoires, également connues sous le nom de random forests, sont des algorithmes d'apprentissage supervisé utilisés principalement pour les tâches de classification et de régression. Elles sont construites en combinant plusieurs arbres de décision indépendants dans un ensemble, chacun entraîné sur un sous-ensemble aléatoire des données d'entraînement et en utilisant un sous-ensemble aléatoire des caractéristiques disponibles [22]. Dans une forêt aléatoire, chaque arbre de décision produit une prédiction indépendante. Ces prédictions sont ensuite combinées pour former une prédiction finale comme le montre la figure 1.11.

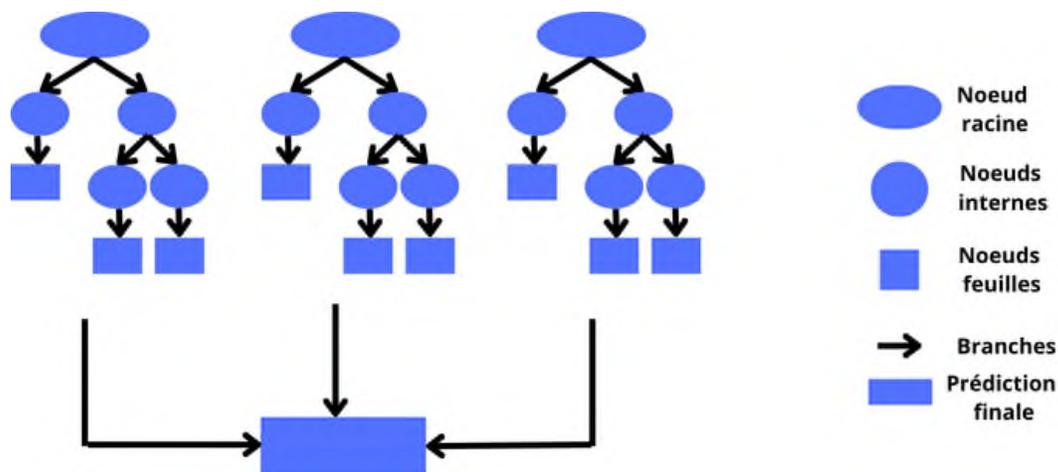


FIGURE 1.11 – Structure d'une forêt aléatoire avec combinaison directe des prédictions.

Les forêts aléatoires sont reconnues pour leur capacité à réduire le surajustement par rapport aux arbres de décision individuels, tout en offrant une précision prédictive élevée. De plus, elles fournissent une estimation de l'importance des caractéristiques, ce qui permet d'identifier les variables les plus influentes dans le processus de prise de décision. En somme, les forêts aléatoires représentent des outils puissants et polyvalents pour la modélisation de données et l'apprentissage automatique, trouvant des applications dans divers domaines [22].

### 1.3.1.4 Régression logistique

La régression logistique (RL) est une technique statistique et d'apprentissage automatique utilisée pour modéliser la probabilité qu'un événement binaire se produise. Cet événement binaire peut être la présence ou l'absence d'une caractéristique, codée par 0 ou 1. Elle est particulièrement utile pour les problèmes de classification binaire, où l'objectif est de classer les observations en deux catégories distinctes [10].

## 1.4 Conclusion

Ce chapitre a présenté les concepts clés des systèmes intelligents, en se focalisant sur les Systèmes de Transport Intelligent (STI). Nous avons discuté des technologies sous-jacentes, de leurs objectifs et défis. Ensuite, l'intelligence artificielle (IA), l'apprentissage automatique et ses différents types ont été introduits. Les STI visent à améliorer la sécurité routière, la fluidité du trafic et à réduire les émissions polluantes grâce à l'utilisation de capteurs avancés et de puissants outils de calcul. Cependant, des défis tels que les accidents, la congestion urbaine et l'impact environnemental persistent et nécessitent des solutions innovantes et efficaces.

Dans ce mémoire, nous nous concentrons sur deux techniques spécifiques de l'apprentissage automatique : les réseaux de neurones récurrents à mémoire à long terme (LSTM) et les forêts aléatoires (RF). Les LSTM sont particulièrement adaptés à la gestion des séquences temporelles et permettent de capturer les dépendances à long terme dans les données, rendant ainsi les prévisions plus précises. Les forêts aléatoires, quant à elles, sont résistantes au surajustement et fournissent des prédictions fiables tout en offrant une interprétabilité des décisions. Ces deux méthodes combinent des capacités analytiques avancées pour traiter de grandes quantités de données et faciliter la prise de décision dans les STI, contribuant ainsi à créer des systèmes de transport plus intelligents et adaptatifs.

# 2

## Etat de l'art sur les comportements dangereux de conduite

### 2.1 Introduction

La sécurité routière est devenue une préoccupation à l'échelle mondiale, alors que les comportements dangereux de conduite continuent de causer un nombre alarmant d'accidents et de pertes de vie. Face à ce défi, de nombreux chercheurs scientifiques ont intensifié leurs efforts pour proposer de nouvelles découvertes visant à améliorer la sécurité sur les routes. Les avancées technologiques, ont considérablement transformé les techniques de détection des comportements anormaux de conduite, ouvrant ainsi la voie à des solutions plus efficaces et innovantes

Dans cette section, nous passons en revue quelques travaux existants qui portent ce domaine, en mettant l'accent sur les méthodes utilisées pour détecter ces comportements à risque, en analysant leurs différences et une comparaison en fonction des critères établis.

### 2.2 Classification des études précédentes

La classification des travaux précédents est une démarche essentielle visant à organiser et à synthétiser les travaux de recherche antérieurs. Cette approche permet de mettre en lumière les différentes perspectives, méthodologies et résultats des études précédentes, facilitant ainsi une compréhension globale des comportements anormaux de conduite. Elle servira également de base solide pour positionner les nouvelles contributions et identifier les avenues de recherche prometteuses. Voici notre schéma de classification pour ces travaux, illustré dans la figure 2.1.

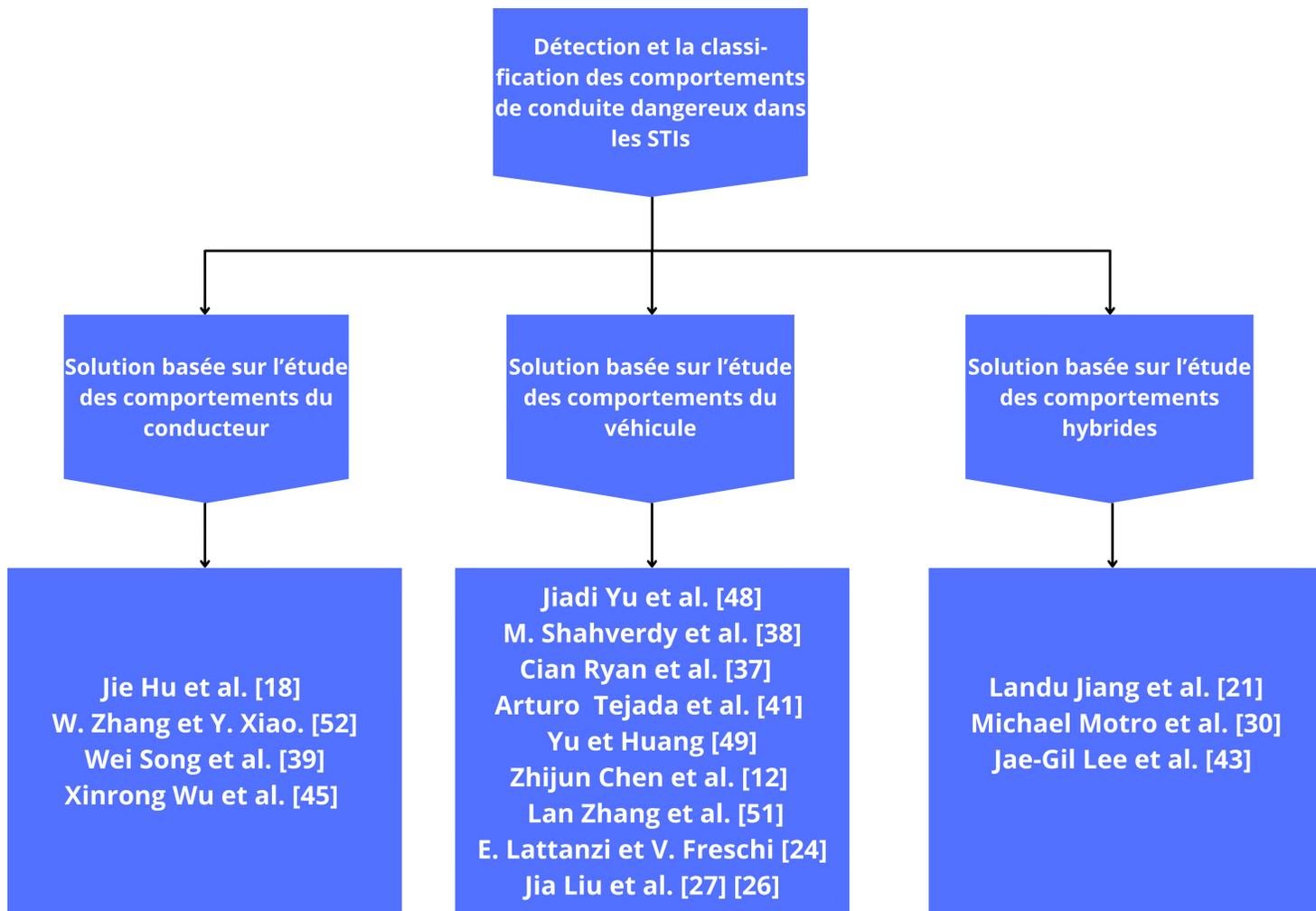


FIGURE 2.1 – Classification des travaux passés en revue.

## 2.3 Étude des travaux existants sur la détection et la classification des comportements de conduite dangereux dans les STIs

Les comportements dangereux de conduite sont l'un des défis de recherche en matière de sécurité routière dans les STIs. Quelques études ont démontré que ces comportements anormaux sont causés par des interactions entre les conducteurs, les véhicules ou les deux à la fois.

### 2.3.1 Solution basée sur l'étude des comportements du conducteur

Les réactions du conducteur jouent un rôle très important dans la prévention des accidents et la réduction des risques sur les routes, une conduite responsable et attentive demeure la clé pour assurer la sécurité de tous les usagers de la route. Les facteurs tels que l'attention, la vigilance, le respect des règles de circulation sont autant d'éléments influencés par le conducteur et qui ont un impact direct sur la sécurité routière.

- **Abnormal Driving Detection Based on Normalized Driving Behavior.**

Jie Hu et al. [18] ont proposé une méthode de détection de la conduite anormale en analysant le comportement de conduite normalisé. Face au problème de la sécurité routière, l'étude vise à utiliser des capteurs embarqués dans les véhicules. Ils ont proposé d'évaluer quantitativement les styles de conduite en normalisant le comportement de conduite sur la base de la modélisation du conducteur personnalisée, qui sert de base à la détection de la conduite anormale dans cette étude. Le comportement de conduite brut est normalisé en utilisant le conducteur virtuel ou le modèle de conducteur personnalisé pour effectuer la tâche de suivi de vitesse telle que définie par le test du cycle de conduite standard, par exemple, le FTP-72. Le modèle de conducteur personnalisé est établi en utilisant un réseau neuronal conçu localement et des données d'essai de véhicules du monde réel (VTD), où seuls la vitesse du véhicule (VS), la position du papillon des gaz (TP) et la pression de freinage (BP) sont utilisés pour simplifier. En raison du manque de données réelles sur la conduite anormale, trois comportements anormaux typiques sont caractérisés et simulés pour la détection, à savoir la conduite fatiguée/en état d'ivresse, la conduite imprudente et l'utilisation du téléphone en conduisant. De plus, un indice d'anormalité ( $SSE, \sigma_{TP}, \sigma_{BP}, \sigma_{*TP}$ ) est proposé pour évaluer le comportement de conduite anormal. L'étude inclut une analyse statistique des comportements de conduite anormaux, avec des exemples de résultats pour différents conducteurs et situations. Ces indices et résultats quantitatifs fournissent une évaluation approfondie de l'efficacité de la méthode de détection de conduite anormale proposée.

- **Detection of Dangerous Driving Behavior Based on CG-YOLOv5.**

Weiguo Zhang et Yunxia Xiao. [52] présentent une avancée significative dans la détection des comportements de conduite dangereuse en temps réel, mettant l'accent sur la problématique croissante des accidents liés à l'utilisation du téléphone portable pendant la conduite en Chine. Face à ce défi en matière de sécurité routière, leur objectif est de perfectionner la détection tout en atténuant les interférences de bruit des caractéristiques de fond, en se concentrant particulièrement sur l'identification des comportements à risque. La méthode améliorée proposée est appelée CG-YOLOv5 se distingue par l'intégration du module d'attention de bloc de convolution (CBAM) qui joue un rôle crucial en réduisant les interférences de bruit des caractéristiques de fond et l'adoption de la convolution Ghost qui remplace la convolution ordinaire, permettant ainsi de réduire la taille du modèle sans sacrifier la performance. Ces ajouts visent à renforcer l'efficacité de détection tout en réduisant la complexité du modèle, offrant ainsi une solution plus efficace et moins gourmande en ressources. Les expérimentations menées dans un environnement défini, avec des jeux de données variés, mettent en lumière les indices d'évaluation tels que la précision, le rappel, la mAP, et le FPS. Les résultats obtenus démontrent une amélioration par rapport à la méthode de détection du réseau d'origine. Cette amélioration se manifeste par une augmentation de la précision, du rappel et une réduction du temps de traitement, renforçant ainsi l'applicabilité de la méthode proposée dans la détection des comportements de conduite dangereux.

- **Identification of dangerous driving state based on lightweight deep learning model.**

Wei Song et al. [39] se penchent sur l'identification de l'état de conduite dangereux en utilisant un modèle d'apprentissage profond léger. Ils commencent par présenter leur approche, qui repose sur l'utilisation d'un réseau de neurones convolutif (CNN) pour extraire les caractéristiques des

images de conduite capturée à partir des caméras embarquées. Ces caractéristiques comprennent des informations sur le comportement du conducteur, telles que la présence de téléphone portable ou de cigarette, le non-respect des feux de signalisation, et d'autres comportements dangereux. Dans le processus de prétraitement des données, les images de conduite sont collectées et étiquetées pour former un ensemble de données d'entraînement. Les auteurs expliquent également comment ils ont utilisé un modèle d'apprentissage en profondeur léger Shuffle Net, spécifiquement conçu pour les applications embarquées, pour entraîner leur système à reconnaître les comportements de conduite dangereux en lui apportant quelques améliorations spécifiques pour mieux répondre aux exigences de la tâche de classification des comportements de conduite dangereux. Cela comprend notamment des modifications de la structure du réseau en utilisant la technique de segmentation des canaux et mélange de canaux, l'optimisation des fonctions d'activation en remplaçant par exemple la fonction ReLU par la fonction LeakyReLU, et des techniques de visualisation des poids de classification telles que les Heatmaps, le Guided-Backpropagation et le Grad-CAM. Les résultats de leur approche montrent que leur modèle parvient à identifier avec précision les comportements de conduite dangereux dans une variété de situations. Les auteurs discutent également des implications de leur recherche et des possibilités d'amélioration future, telles que l'intégration de données en temps réel provenant de capteurs embarqués dans les véhicules, sinon ils soulignent l'importance de leur travail pour améliorer la sécurité routière et la conception de systèmes d'assistance à la conduite.

- **Abnormal Driving Behavior Detection for Bus Based on the Bayesian Classifier**

Xinrong Wu et al. [45] ont présenté une méthode pour la détection et l'identification des comportements de conduite anormaux des autobus, soulignant l'importance de la sécurité routière et la réduction des accidents. Ils ont construit un ensemble de données annotées en utilisant des capteurs de smartphones embarqués dans des autobus réels pour collecter des données de conduite, mettant l'accent sur l'utilisation de l'accéléromètre et du capteur d'orientation pour obtenir des informations telles que l'accélération, l'orientation et le timestamp. Les étapes comprennent la collecte des données, puis l'entraînement du classificateur naïf de Bayes à partir de ces données collectées, qui comprennent des informations telles que l'accélération, l'orientation et le timestamp des comportements de conduite. Ensuite, ils ont extrait des caractéristiques pertinentes à partir de ces données, comprenant des mesures telles que la variation de l'accélération, l'écart type de l'accélération, la variation de l'orientation, etc. Dans cette étude, la classification s'est déroulée comme suit : Le classificateur naïf de Bayes commence par calculer les probabilités a priori de chaque classe, ce qui signifie qu'il calcule la probabilité qu'un comportement de conduite appartienne à chaque classe. Ensuite, il calcule les probabilités conditionnelles de chaque classe étant donné les caractéristiques observées, en supposant que ces caractéristiques sont conditionnellement indépendantes les unes des autres. Enfin, le classificateur utilise le théorème de Bayes pour combiner les probabilités a priori et conditionnelles afin de calculer la probabilité finale que le comportement de conduite appartienne à chaque classe. Il choisit ensuite la classe avec la probabilité la plus élevée comme prédiction pour le comportement de conduite observé, tels que le freinage soudain, le changement de voie nonchalant, le virage rapide, le demi-tour rapide et le stationnement prolongé. Ils ont comparé ce classificateur (NB) avec les machines à vecteurs de support (SVM). Les résultats expérimentaux mettent en évidence l'efficacité de la méthode proposée, avec une précision atteignant 98,40% grâce au classificateur bayésien naïf, surpassant la performance de la SVM évaluée à 96,89%. Les comparaisons des résultats ont été effectuées à travers des mesures telles

que l'exactitude, la précision, le rappel et le taux de faux positif, renforçant ainsi la crédibilité des conclusions.

### 2.3.2 Solution basée sur l'étude des comportements du véhicule

Les solutions basées sur le comportement des véhicules utilisent des données en temps réel telles que la vitesse, la position et les mouvements des véhicules pour optimiser la navigation et améliorer la sécurité. Ces technologies intègrent l'IA pour fournir des systèmes de transport intelligents, des assistants de conduite avancés et des véhicules autonomes, contribuant ainsi à une mobilité plus sûre, plus efficace et plus respectueuse de l'environnement. Dans cette partie, nous parlerons des travaux qui sont basés sur ce concept.

- **Fine-Grained Abnormal Driving Behaviors Detection and Identification with Smartphones.**

Dans ce travail [48], Jiadi Yu et al. ont introduit une approche conçue pour la détection et l'identification fine des comportements de conduite anormaux en exploitant les capacités des smartphones en utilisant le système de D3 comme une application Android installée sur ces derniers, ce système fonctionne en plusieurs étapes. Tout d'abord, il collecte des données d'accélération et d'orientation à partir des capteurs des smartphones des conducteurs pendant la conduite réelle. Ensuite, ces données sont traitées pour identifier les comportements de conduite anormaux tels que le tissage, les changements de voie brusques, le dérapage, les demi-tours rapides, les virages à large rayon et les freinages brusques. Une fois que les données sont prétraitées, D3 utilise les informations recueillies pour entraîner des modèles de classification, notamment des machines à vecteurs de support (SVM) et des réseaux neuronaux (RNN), afin de distinguer les comportements de conduite anormaux des comportements normaux. Une fois que les modèles sont entraînés, D3 les applique pour identifier en temps réel les comportements de conduite anormaux dans les données collectées. Les comportements détectés peuvent être enregistrés ou signalés aux conducteurs. D3 évalue ses performances en comparant les comportements identifiés avec des données de référence ou des annotations manuelles, ce qui permet d'améliorer la précision et la fiabilité du système. Les résultats des expériences ont révélé que le système proposé a obtenu des performances remarquables dans la détection et l'identification des comportements de conduite anormaux. Les mesures d'évaluation telles que la précision, le rappel et le taux de faux positifs ont démontré la capacité de D3 à discriminer efficacement les comportements anormaux des comportements normaux avec une précision élevée. De plus, l'article a analysé l'impact de divers facteurs tels que la taille de l'ensemble de données, les conditions de circulation, le type de route et le placement du smartphone sur les performances de D3. Cependant, ils ont également souligné certaines limites, notamment la dépendance aux capteurs de smartphone, la sensibilité aux conditions environnementales et les contraintes liées à la taille de l'ensemble de données. Ils ont formulé des recommandations pour de futures recherches, notamment l'amélioration de la robustesse du système dans des conditions variables et l'exploration de nouvelles techniques d'apprentissage automatique pour améliorer encore les performances.

- **Driver behavior detection and classification using deep convolutional neural networks.**

Mohammad Shahverdy et al. [38] ont employé une approche multifacette pour la détection des comportements de conduite. Tout d'abord, ils ont utilisé des capteurs embarqués dans le véhicule

pour recueillir une gamme de données, notamment l'accélération, la gravité, les tours par minute (RPM), la vitesse et la position de la pédale d'accélérateur. Ensuite, ils ont appliqué la technique du tracé de récurrence afin de transformer les dépendances temporelles des signaux de conduite en dépendances spatiales. Cette méthode a permis de représenter les données de conduite sous forme d'images, facilitant ainsi leur traitement par les réseaux neuronaux convolutionnels (CNN). Les chercheurs ont ensuite utilisé des architectures de CNN pour classifier les comportements de conduite à partir des images ainsi générées, en variant le nombre de couches convolutionnelles, le nombre de filtres dans chaque couche et la taille des filtres pour évaluer différentes configurations, différentes architectures de CNN ont été évaluées pour trouver un compromis entre précision et complexité computationnelle. Le modèle optimal a été identifié comme ayant deux couches convolutionnelles, 16 filtres dans la première couche et une taille de filtre de  $7 \times 7$ . Pour l'apprentissage et la classification des comportements de conduite, les chercheurs ont exploité comme l'une des principales bibliothèques pour implémenter et entraîner les modèles de CNN et la bibliothèque Keras en Python. Ils ont entraîné les modèles de CNN sur les données collectées et ont évalué leurs performances en termes de précision et de complexité computationnelle, offrant ainsi une solution complète et idéale pour la détection des comportements de conduite. Les résultats ont montré une précision élevée de 99,98 % pour le modèle de référence, avec une complexité faible. Les chercheurs ont constaté l'impact de la taille et du nombre de filtres dans les CNN sur la précision du modèle.

- **End-to-End Autonomous Driving Risk Analysis : A Behavioural Anomaly Detection Approach.**

La problématique étudiée [37] concerne l'évaluation de la sécurité des véhicules autonomes (AV) par rapport aux conducteurs humains, en utilisant des méthodes proactives basées sur le comportement de conduite. Cian Ryan et al. cherchent à déterminer si les AV sont au moins aussi sûrs que les conducteurs humains moyens, voire plus sûrs, afin de faciliter leur déploiement et d'encourager l'acceptation sociétale de cette nouvelle technologie. Les capteurs utilisés pour recueillir les données d'entrée comprennent généralement des caméras, des capteurs lidar, des capteurs radar et des capteurs de vitesse pour le but d'étudier des types de comportements anormaux incluent des actions telles que le freinage brusque, l'accélération, les changements de voie brusques, la vitesse excessive ou insuffisante, etc. L'étude faite comprennent l'utilisation de réseaux de neurones convolutionnels (CNN) en utilisant l'algorithme de rétropropagation du gradient pour l'entraînement des modèles de prédiction de commandes de direction et de vitesse des AV, les CNN sont composés de plusieurs couches, y compris des couches de convolution et des couches entièrement connectées. Les chercheurs utilisent une architecture spécifique proposée par Bojarski et al. (référence 46 dans l'article) comprenant 9 couches au total, dont des couches de normalisation, 5 couches de convolution et trois couches entièrement connectées. Ainsi que l'utilisation de processus gaussiens (GP), ils utilisent des statistiques pour identifier les schémas de conduite dangereux ou inattendus. Ils modélisent les distributions de probabilité des comportements de conduite normaux à l'aide de ce GP, puis identifient les comportements qui s'écartent de ces distributions de manière significative. Les résultats comprennent des scores de risque calculés pour les AV et les conducteurs humains sur différents parcours de test, ainsi que des analyses détaillées des comportements de conduite anormaux détectés. Les chercheurs identifient des schémas de conduite dangereux ou inattendus et proposent des pistes pour améliorer la sécurité des AV.

- **Towards a Characterization of Safe Driving Behavior for Automated Vehicles Based on Models of “Typical” Human Driving Behavior.**

La problématique étudiée par Arturo Tejada et al. [41] dans cette approche concerne la caractérisation du comportement de conduite sûr pour les véhicules automatisés, en particulier en ce qui concerne les interactions de conduite typiques dans un contexte de trafic mixte. Ils cherchent à déterminer comment les véhicules automatisés peuvent être programmés pour conduire de manière sûre et prévisible, en imitant le comportement humain typique sur la route. Les méthodes proposées dans ce travail reposent principalement sur l'utilisation d'autoencodeurs, une architecture de réseau de neurones non supervisée utilisée pour l'encodage et le décodage de données, pour modéliser ces interactions de conduite typiques et détecter les anomalies dans les données de conduite naturaliste. Les types de comportements étudiés sont des interactions de conduite longitudinales entre deux véhicules, telles que les dépassements, les changements de voie, et les ralentissements. Les chercheurs se sont concentrés sur l'utilisation de capteurs de données provenant de caméras vidéo stationnaires, qui ont enregistré les trajectoires des véhicules à une fréquence de 10 Hz. Les données capturées comprennent les indicateurs de vitesse, d'accélération, de position latérale, de distance et de temps de collision entre les véhicules, un filtre de Savitzky-Golay est utilisé pour lisser les signaux de vitesse et d'accélération des véhicules dans les données de conduite normal afin de réduire le bruit et d'améliorer la qualité de ces données. Les autoencodeurs sont formés à partir de données de conduite naturaliste extraites de l'ensemble de données NGSIM. Quelques détails d'implémentation des autoencodeurs sont mentionnés, y compris sa structure, Les fonctions d'activation telles que ReLU (Rectified Linear Units) sont utilisées dans les couches cachées des autoencodeurs pour introduire une non-linéarité dans le modèle et les fonctions sigmoïdes sont utilisées dans la couche de sortie pour produire des reconstructions des indicateurs de conduite, et l'algorithme d'optimisation (Adam). En outre, une méthodologie de détection d'anomalies basée sur les autoencodeurs est décrite, où les interactions de conduite atypiques sont identifiées en comparant les reconstructions des indicateurs de conduite avec les données d'entrée originales. Ils ont dû définir des seuils pour déterminer à quel point une erreur de reconstruction est considérée comme suffisamment importante pour étiqueter une interaction de conduite comme anormale. Les chercheurs ont constaté que la plupart des interactions anormales étaient des valeurs aberrantes avec une gravité plus élevée que les interactions typiques. De plus, ils ont observé une corrélation entre les interactions anormales détectées à différentes tailles de fenêtre, et ont souligné l'importance de prendre en compte plusieurs tailles de fenêtre pour détecter les anomalies efficacement.

- **A deep encoder-decoder network for anomaly detection in driving trajectory behavior under spatio-temporal context.**

Le travail de Yu et Huang [49] a apporté une contribution dans le domaine de la détection des comportements anormaux de conduite. Les auteurs proposent un cadre novateur nommé STDTB-AD basé sur un réseau encodeur-décodeur pour évaluer le degré d'anormalité des conducteurs. L'approche se divise en trois modules clés : prétraitement des données, modélisation des trajectoires et détection d'anomalies. Les données des trajectoires de conduite sont prétraitées en utilisant des techniques de cartographie et de nettoyage, cette étape vise à garantir la qualité des données entrantes, en éliminant les erreurs potentielles et en créant ainsi une base solide pour les phases ultérieures du processus. La modélisation des trajectoires prend en compte le contexte spatial et temporel, La partition du réseau routier en unités spatio-temporelles homogènes permet de

capturer la complexité des comportements de conduite dans différentes situations, cette approche reflète la réalité des mouvements des conducteurs, dépassant les approches plus simplistes qui ignorent l'aspect spatio-temporel, la détection d'anomalies repose sur un modèle de variational auto-encoder (VAE) formé pour comprendre les motifs de comportement de conduite dans divers contextes spatio-temporels. Les évaluations expérimentales utilisent des données simulées et réelles de trajets de taxi à Beijing, un choix pertinent pour la diversité des scénarios de conduite. STDTB-AD est soumis à une batterie de comparaisons avec des méthodes traditionnelles telles que Facteur Local d'Anomalie (LOF), Forêt d'Isolation (iForest), Machines à Vecteurs de Support à une Seule Classe (OC-SVM) et TPRRO ainsi qu'avec des approches qui ne considèrent que le contexte spatial ou temporel. Les résultats montrent que STDTB-AD offre une approche complète et novatrice pour la détection des comportements anormaux de conduite, combinant avec succès le prétraitement des données, la modélisation spatio-temporelle et la détection d'anomalies par VAE et surpasse les méthodes traditionnelles en termes de précision, de rappel et de F1-score.

- **Dangerous driving behavior detection using video-extracted vehicle trajectory histograms.**

Zhijun Chen et al. [12] étudiées une problématique qui concerne la détection des comportements de conduite dangereux à partir de vidéos de surveillance routière. Les chercheurs se sont intéressés à développer une méthode efficace pour détecter ces comportements à partir des données vidéo collectées par des caméras vidéo installées sur des bâtiments en bordure de route pour capturer le trafic routier, pour classifier les comportements de conduite anormaux, notamment le changement fréquent de voie (FLC), changement de voie abrupt (ACL), conduite rétrograde (RD), occupation illégale de voie (ILO) et demi-tour illégal (IT). Les chercheurs proposent des méthodes qui commencent par l'extraction des caractéristiques à partir des vidéos originales telle que les caractéristiques les plus informatives pour la détection de ces comportements anormaux ont été identifiées comme étant la variation d'angle entre deux angles de contrôle adjacents (H5) et la vitesse moyenne parmi les points de trajectoire entre deux angles de contrôle adjacents (H7), sélectionnées à l'aide de l'algorithme mRMR, ensuite la construction d'un modèle de détection utilisant la méthode SVM avec une optimisation des paramètres via PSO (Particle Swarm Optimization), l'évaluation des caractéristiques à l'aide d'algorithmes comme mRMR comparés à CMIM, MIM et Relief, ainsi les performances de la méthode proposée qui combine l'algorithme PSO-SVM avec des histogrammes de trajectoire. Les résultats attendus sont les prédictions du modèle de détection concernant les comportements de conduite dangereux. Les chercheurs mettent en évidence l'efficacité de PSO-SVM, basée sur l'utilisation d'histogrammes de trajectoire par rapport à d'autres méthodes évaluées telles que NBC, kNN et C4.5 decision tree. Ils identifient également les caractéristiques les plus informatives pour la détection des comportements de conduite dangereux.

- **A Machine Learning Based Defensive Alerting System Against Reckless Driving in Vehicular Networks.**

Les recherches faites par Lan Zhang et al. [51] portent sur la détection proactive des comportements de conduite imprudents sur la route, en particulier la conduite dangereuse, et la notification préventive des menaces potentielles que représentent les véhicules imprudents approchants. Pour répondre à cette problématique, les chercheurs ont développé un système d'alerte défensif qui utilise des capteurs tels que les dispositifs de mesure de la sécurité routière (SSM) pour détecter les rencontres conflictuelles et enregistrer les valeurs correspondantes. Les données d'entrée

comprennent des caractéristiques de conduite telles que la durée de l'interaction avec un autre véhicule, l'écart de distance, la vitesse relative, ainsi que les données capturées par SSM enregistrant les rencontres conflictuelles. Les méthodes proposées comprennent d'abord une évaluation théorique de la probabilité de collision des manœuvres typiques de conduite imprudente, comme les changements de voie imprudents, afin de comprendre les conséquences potentielles sur la sécurité routière. Ensuite, un mécanisme de notation des performances de conduite basé sur l'apprentissage automatique (CDPR) est développé, exploitant les capacités de calcul des véhicules voisins et d'un serveur cloud. Des algorithmes d'apprentissage automatique tels que SVM et l'arbre de décision sont utilisés pour évaluer les performances de conduite en classifiant les conducteurs en fonction de leur comportement. Enfin, un simulateur de trafic appelé Simulation of Urban MObility (SUMO) est utilisé pour générer des données de comportement de conduite, essentielles pour entraîner les modèles de notation des performances de conduite. Les résultats montrent que le système d'alerte défensif peut détecter avec précision les véhicules imprudents, et les algorithmes d'apprentissage automatique, en particulier le modèle d'arbre de décision, ont montré des performances élevées dans la notation des performances de conduite. Les données de sortie incluent la notation des performances de conduite de chaque véhicule ainsi que les alertes générées en cas de conduite imprudente détectée.

- **Machine Learning Techniques to Identify Unsafe Driving Behavior by Means of In-Vehicle Sensor Data.**

Dans cette étude de Emanuele Lattanzi et Valerio Freschi [24] se penchent sur la problématique de la reconnaissance des comportements de conduite sûrs et dangereux à partir des données collectées par les capteurs intégrés dans les véhicules. Leur objectif est de développer une méthodologie efficace utilisant des techniques d'apprentissage automatique pour identifier ces comportements à partir des données disponibles. Pour ce faire, ils ont recours à deux algorithmes de classification :

1. Un SVM (Machine à Vecteurs de Support) : utilisé pour la classification des comportements de conduite. Il utilise pour entraîner le modèle à distinguer entre les comportements sûrs et dangereux en se basant sur les données des capteurs.
2. Un réseau neuronal à propagation avant : également utilisé pour la classification. Cet algorithme utilise une méthode d'optimisation qui est la rétropropagation Levenberg-Marquardt utilisée pour entraîner les poids d'un réseau neuronal en ajustant itérativement les valeurs des poids afin de minimiser une fonction de perte.

Ces algorithmes sont entraînés et testés sur un ensemble de données comprenant à la fois des signaux OBD-II standard et des données non standard telles que la pression du système de freinage et l'angle de direction. Les comportements étudiés incluent ceux qui sont identifiés à partir des données de mouvement du véhicule. Les données d'entrée sont les signaux captés par ces capteurs, tandis que les données de sortie sont les classifications des comportements de conduite en tant que sûrs ou dangereux. Les résultats obtenus démontrent une précision moyenne de plus de 90% pour les deux classificateurs, avec un léger avantage pour le réseau neuronal.

- **Symbolic aggregate approximation based data fusion model for dangerous driving behavior detection.**

Jia Liu et al. [27] présentent une nouvelle approche pour détecter les comportements de conduite dangereux à partir de données de trajectoire de véhicules en prenant au compte les conditions

environnementales. Ils commencent par exposer le problème de la détection des comportements de conduite dangereux, soulignant l'importance de cette tâche pour la sécurité routière. Ensuite, ils proposent le modèle SAX-DF, qui combine la technique Symbolic Aggregate approXimation (SAX) et l'apprentissage profond (Deep Learning) pour analyser les données de trajectoire des véhicules, ainsi que les données météorologiques pour mieux comprendre les conditions environnementales lors des événements de conduite dangereux. Le modèle SAX-DF se compose de plusieurs étapes. Tout d'abord, les données de trajectoire sont prétraitées à l'aide de la méthode SAX pour les représenter sous forme de symboles. Ensuite, un réseau de neurones profond est utilisé pour extraire les caractéristiques des séquences de symboles. Ces caractéristiques sont ensuite fournies à un classificateur (n'est pas mentionné) pour détecter les comportements de conduite dangereux. La dernière étape consiste à intégrer les conditions météorologiques sous 3 étapes

1. Collecte des données externes capturées.
2. Associer ces données avec les données de conduites par exemple " si y avait de la pluie la route glisse du coup on aura des tissages ou des zigzags".
3. Analyse les interactions en utilisant le modèle de SAX-DF entre les données environnementales et celle de conduite.

Les chercheurs détaillent également les différentes métriques d'évaluation utilisées pour évaluer les performances de leur modèle, telles que le rappel, la précision et la mesure F et décrivent en détail les ensembles de données utilisés pour évaluer le modèle SAX-DF, ainsi que les différentes méthodes de comparaison utilisées dans l'étude, notamment les algorithmes de classification SVM, NBC(Naive Bayesian Classifier) et GBC(Gaussian Bayesian Classifier), ils ont utilisé aussi des modèles de graphiques GM pour éclaircir leurs travaux. Ils présentent ensuite les résultats de l'évaluation, mettant en évidence les performances de leur modèle par rapport aux autres méthodes de détection des comportements de conduite dangereux.

- **SLAFusion : Attention fusion based on SAX and LSTM for dangerous driving behavior detection.**

Une autre approche [26] qui intègre les conditions météorologiques et qui vise à améliorer quelques défis de travail précédents qui est la perte d'information due à la segmentation des processus de conduite [27] pour mieux détecter les comportements de conduite dangereux en combinant les caractéristiques des processus de conduite de différentes granularités et en intégrant les informations environnementales avec l'utilisation des données des capteurs du véhicule, telles que la vitesse, l'accélération, le taux angulaire, etc., sont collectées pendant la conduite. Jia Liu et al. introduisent d'abord le concept de fusion d'attention, qui utilise des mécanismes d'attention pour évaluer l'importance relative des différentes caractéristiques lors de la détection des comportements dangereux classifiés en 2 sous-classes :

1. Mécanisme d'auto-attention (self-attention) : Il repose sur le calcul de la corrélation entre les caractéristiques, ce qui permet de déterminer celles qui sont les plus pertinentes pour la détection des comportements dangereux.
2. Mécanisme de fusion des informations auxiliaires : Il utilise un réseau de neurones simple pour extraire les caractéristiques des informations environnementales, qui sont ensuite combinées avec les caractéristiques des processus de conduite à l'aide du mécanisme d'auto-attention.

Ensuite, ils détaillent la méthode SLAFusion, qui associe l'approximation d'agrégation de segmentation pour modéliser les changements d'état du véhicule avec une fusion d'attention des caracté-

ristiques extraites des processus de conduite, notamment via l'utilisation de la transformée symbolique approximative (SAX) et des réseaux de neurones récurrents (LSTM) utiliser pour éviter le problème de perte de donnée. Pour évaluer leur méthode, Jia Liu et al. mènent des expériences sur des ensembles de données réelles, comparant les performances de SLAFusion avec plusieurs méthodes basées sur des baselines traditionnels et des approches récentes telles que le SVM(Support Vector Machine), NBC(Naive Bayesian Classifier), GBC(Gaussian Bayesian Classifier),etc.Les résultats démontrent que la méthode proposée surpasse les autres méthodes en termes de rappel et de score F-score.

### 2.3.3 Solution basée sur l'étude des comportements hybrides

Ces comportements émergent lorsque les actions du conducteur et les réponses du véhicule se combinent de manière imprévisible, conduisant à des situations potentiellement dangereuses. Par exemple, un conducteur peut effectuer une manœuvre brusque en réponse à une alerte de collision émise par le système de freinage automatique du véhicule. Identifier et comprendre ces comportements sont essentiels pour améliorer les systèmes de sécurité automobile et prévenir les accidents. Cela nécessite une analyse approfondie des interactions entre le conducteur et le véhicule.

- **Smart Diagnosis : Deep Learning Boosted Driver Inattention Detection and Abnormal Driving Prediction.**

Landu Jiang et al. [21] abordent la détection de la conduite anormale en utilisant des données de trajectoire de véhicules réels fournies par une compagnie d'assurance commerciale, au lieu d'utiliser un simulateur spécifique pour leurs expérimentations. Les capteurs utilisés comprennent les dispositifs OBD-II et les caméras pour enregistrer les dynamiques de conduite, ainsi que les capteurs IMU (accéléromètre et gyroscope) pour une extension du système en temps réel. Ils définissent d'abord le problème et proposent une méthodologie basée sur l'apprentissage profond pour identifier les comportements inattentifs des conducteurs et prédire les opérations de conduite anormales, telles que l'accélération ou la décélération anormales, ainsi que les changements de voie. Pour la détection des événements non sécuritaires, ils comparent plusieurs algorithmes d'apprentissage automatique tels que les arbres de décision boostés par gradient, les machines à vecteurs de support (SVM), les voisins les plus proches (KNN) et la régression linéaire (LR). Ils démontrent que le modèle de forêt aléatoire (RF) surpasse les autres avec une précision, un rappel et un score F1 élevés. Pour la reconnaissance de la conduite inattentive, ils proposent l'utilisation d'un modèle de réseau de neurones convolutif profond (Inception v3) avec la technique Mixup, comparant sa performance à celle de méthodes classiques comme SVM, GBDT, CNN et DNN. Le modèle Inception v3 avec Mixup surpasse les autres méthodes avec une meilleure précision et un score F1 plus élevé. Enfin, pour la prédiction des opérations de conduite anormales, ils utilisent un modèle basé sur les réseaux de neurones récurrents LSTM, comparant sa performance à celle de modèles RNN et SVM. Le modèle LSTM surpasse les autres en termes de précision.

- **Vehicular ad-hoc network simulations of overtaking maneuvers on two-lane rural highways.**

Michael Motro et al. [30] ont concentré dans cette étude sur la modélisation des dépassements sur les routes rurales à deux voies en utilisant des simulations de réseaux ad hoc véhiculaires. Elle débute par une contextualisation des enjeux de sécurité routière sur ces routes, souvent associées

à un nombre disproportionné d'accidents mortels lors de dépassements. Ils ont développé un système de détection de collision DSRC (Dedicated Short-Range Communication) visant à anticiper et à éviter les collisions potentielles lors de ces manœuvres. Leur approche repose sur le modèle de simulation du trafic routier "Vehicles in Network Simulation" (Veins), prenant en compte divers paramètres des véhicules, le comportement des conducteurs et les performances du système DSRC. L'article présente une analyse détaillée des résultats des simulations, mettant en lumière l'influence de facteurs tels que le temps de réaction des conducteurs, les vitesses et accélérations des véhicules, ainsi que les paramètres de communication DSRC. Les auteurs utilisent un modèle de classification binaire qui est la régression logistique pour évaluer l'impact de ces facteurs sur les collisions, la détection des collisions et les faux avertissements émis par le système DSRC. Les résultats démontrent une capacité du système DSRC à prédire avec succès environ 67% des collisions simulées, avec moins de 4% de faux avertissements pour les dépassements sûrs. Ils soulignent notamment l'importance de la portée de communication DSRC et la tolérance aux erreurs de paquet jusqu'à 50%, ainsi que l'impact limité des erreurs de capteur et d'estimation sur la détection des collisions.

- **DF-TAR : A Deep Fusion Network for Citywide Traffic Accident Risk Prediction with Dangerous Driving Behavior.**

Dans cette étude [43], Jae-Gil Lee et al. abordent la problématique de prédiction des risques d'accidents de la circulation à l'échelle d'une ville, en prenant en compte les comportements de conduite dangereux telles que l'utilisation du téléphone portable en conduisant, la conduite en état d'ébriété, la conduite agressive, les dépassements risqués, le suivi trop proche des autres véhicules. Ils introduisent un réseau de fusion profonde (DF-TAR) qui intègre les comportements de conduite dangereux et diverses données environnementales pour prédire les risques d'accidents dans chaque district d'une ville à des intervalles de temps spécifiques en utilisant différents algorithmes tels que les arbres de décision, la régression linéaire et la méthode des k plus proches voisins KNN. Leur méthodologie commence par quantifier les scores de corrélation entre les comportements de conduite dangereux et les enregistrements d'accidents passés pour justifier leur intégration dans le modèle de prédiction. Le DF-TAR est basé sur des techniques de deep learning, notamment des réseaux de neurones convolutionnels (CNN) et des réseaux de neurones récurrents (RNN) tels que les autoencodeurs débruteurs empilés, les autoencodeurs convolutionnels empilés et les réseaux LSTM (Long Short-Term Memory) pour prédire la fréquence et le risque d'accidents de la circulation à l'échelle de la ville en intégrant des données spatiales et temporelles. Ces algorithmes permettent au modèle de capturer des informations complexes à partir de données hétérogènes et de les fusionner efficacement pour une prédiction précise des risques d'accidents. Les résultats de leur évaluation montrent que la méthode proposée surpasse les modèles de base, avec une amélioration des mesures d'erreur moyenne absolue (MAE) et de l'erreur quadratique moyenne (RMSE). Ils envisagent à la fin l'intégration des comportements de conduite dangereux dans les systèmes de prédiction en temps réel des accidents de la circulation, ainsi que leur utilisation potentielle avec des véhicules autonomes pour améliorer la sécurité routière.

## 2.4 Comparaison des travaux étudiés

### 2.4.1 Critère de comparaison

Nous avons établi certains critères de comparaison pour une évaluation approfondie des différents travaux étudiés. Ces critères comprennent l'objectif, les paramètres d'entrée et de sortie, l'algorithme utilisé, la méthode employée, les métriques d'évaluation et la complexité de calcul.

- **Objectif** L'objectif spécifie si l'approche se concentre principalement sur le conducteur, le véhicule, ou les deux. Une approche axée sur le conducteur peut se concentrer sur la détection de comportements spécifiques du conducteur, tandis qu'une approche axée sur le véhicule peut examiner les caractéristiques techniques du véhicule. Certaines approches peuvent également intégrer une perspective combinée, prenant en compte à la fois le comportement du conducteur et les caractéristiques du véhicule pour une évaluation plus holistique de la sécurité routière.
- **Paramètre D'Entrée** Les paramètres d'entrée représentent les données ou caractéristiques utilisées par le modèle pour effectuer ses prédictions. Cela pourrait inclure des variables telles que la vitesse du véhicule, l'accélération, l'angle du volant, etc.
- **Paramètres de Sortie** Cette catégorie indique ce que le modèle cherche à prédire. Il peut s'agir de comportements spécifiques du conducteur, d'événements de conduite, ou d'autres résultats pertinents pour la sécurité routière.
- **Méthode utilisée** Les modèles utilisés pour catégoriser les données d'entrée en classes spécifiques, permettant ainsi la prise de décision.
- **Métrique de performance** Une mesure utilisée pour évaluer l'efficacité et la précision d'une méthode de détection des comportements de conduite. Ces métriques sont généralement basées sur des indicateurs tels que :
  - Le rappel(Acc) : Se concentre sur la capacité du modèle à capturer toutes les occurrences positives, calculé en divisant le nombre de vrais positifs par la somme des vrais positifs et des faux négatifs.
  - La précision(P) : Se concentre sur la capacité du modèle à éviter les fausses alarmes en ne classant pas à tort les instances négatives comme positives, calculée en divisant le nombre de vrais positifs par la somme des vrais positifs et des faux positifs.
  - Le taux de faux positifs(FPR) : Il représente le taux de cas négatifs incorrectement classés comme positifs par le modèle, calculé en divisant le nombre de faux positifs par la somme des vrais négatifs et des faux positifs.
  - Le score F1 : Est la moyenne harmonique de la précision et du rappel, ce qui donne une mesure équilibrée qui prend en compte à la fois les faux positifs et les faux négatifs.
- **Complexité de calcul** Elle peut être évaluée en termes de temps de calcul nécessaire à l'exécution de l'algorithme, souvent exprimé en fonction de la taille de l'entrée, ainsi qu'en termes d'espace mémoire requis pour stocker les données temporaires et les résultats intermédiaires.

### 2.4.2 Tableau comparatif

Article	Objectif	Paramètres d'entrée	Paramètres de Sortie	Méthode utilisée	Metriques	Complexité
[18] 2017	Conducteur	<ul style="list-style-type: none"> <li>- Vitesse</li> <li>- Pression des freins</li> <li>- Position de l'accélérateur</li> <li>- Mouvement conducteur</li> </ul>	<ul style="list-style-type: none"> <li>- Conduit imprudente</li> <li>- Conduit en état de Fatigue/ Ivresse</li> <li>- Utilisation du Téléphone</li> </ul>	- Modèle de conducteur personnalisé		Moyen
[45] 2018	Conducteur	<ul style="list-style-type: none"> <li>- L'accélération</li> <li>- L'orientation</li> <li>-Le timestamp</li> </ul>	<ul style="list-style-type: none"> <li>- le freinage soudain</li> <li>- Le changement de voie nonchalant</li> <li>- Le virage rapide</li> <li>- Le demi-tour rapide</li> <li>- Le stationnement prolongé</li> </ul>	- Naïves de bayes	<ul style="list-style-type: none"> <li>- P= 94.64%</li> <li>- R=94.04%</li> </ul>	Moyen
[52] 2022	Conducteur	<ul style="list-style-type: none"> <li>- Les images</li> <li>- Les vidéos</li> </ul>	<ul style="list-style-type: none"> <li>- Appel téléphonique</li> <li>- Texte avec les mains</li> </ul>	- CG-YOLOv5	<ul style="list-style-type: none"> <li>- P=99.90%</li> <li>- R=99.99%</li> </ul>	Moyen
[39] 2023	Conducteur	<ul style="list-style-type: none"> <li>- Les images</li> </ul>	<ul style="list-style-type: none"> <li>- Présence de téléphone</li> <li>- Présence de cigarette</li> <li>- Le non-respect des feux de signalisation</li> </ul>	<ul style="list-style-type: none"> <li>- CNN</li> <li>- Modèle d'apprentissage profond léger (ShuffleNet)</li> </ul>	<ul style="list-style-type: none"> <li>- P=82,16%</li> </ul>	Elevé

<p><b>[48] 2017</b></p>	<p>Véhicule</p>	<ul style="list-style-type: none"> <li>- L'accélération</li> <li>- l'orientation du véhicule</li> </ul>	<ul style="list-style-type: none"> <li>- le tissage</li> <li>- les changements de voie brusques</li> <li>- le dérapage</li> <li>- les demi-tours rapides</li> <li>- les virages à large rayon</li> <li>- les freinages brusques</li> </ul>	<ul style="list-style-type: none"> <li>- SVM</li> <li>- Réseaux neuronaux</li> </ul>	<ul style="list-style-type: none"> <li>- P(SVM)= 95,36 %</li> <li>- P(NN)= 96,88 %</li> <li>- R(SVM)= 70%</li> <li>- R(NN)= 85%</li> <li>- FPR= 2%</li> </ul>	<p>Elevé</p>
<p><b>[12] 2017</b></p>	<p>Véhicule</p>	<ul style="list-style-type: none"> <li>- Les vidéos de surveillance routière</li> </ul>	<ul style="list-style-type: none"> <li>- Le changement fréquent de voie</li> <li>- changement de voie abrupt</li> <li>- conduite rétrograde</li> <li>- occupation illégale de voie</li> <li>- demi-tour illégal</li> </ul>	<ul style="list-style-type: none"> <li>- PSO-SVM</li> <li>- mRMR</li> </ul>	<ul style="list-style-type: none"> <li>- P(%)</li> <li>- Normal = 95.1</li> <li>- FLC= 87.6</li> <li>- ACL= 86.3</li> <li>- RD= 84</li> <li>- ILO= 85.2</li> <li>- IT= 100</li> </ul>	<p>Moyen</p>
<p><b>[51] 2019</b></p>	<p>Véhicule</p>	<ul style="list-style-type: none"> <li>- La durée de l'interaction avec un autre véhicule, l'écart de distance, la vitesse relative</li> <li>- les données capturées par SSM</li> </ul>	<ul style="list-style-type: none"> <li>- Les alertes générées en cas de conduite imprudente détectée</li> </ul>	<ul style="list-style-type: none"> <li>- SVM</li> <li>- Arbre de décision</li> </ul>		<p>Très Elevé</p>
<p><b>[38] 2020</b></p>	<p>Véhicule</p>	<ul style="list-style-type: none"> <li>- L'accélération</li> <li>- La gravité</li> <li>- Les tours par minute (RPM)</li> <li>- la vitesse</li> <li>- la position de l'accélérateur</li> </ul>	<ul style="list-style-type: none"> <li>- Conduit agressif</li> <li>- Conduit distrait</li> <li>- Conduit somnolent</li> <li>- Conduit en état d'ébriété</li> </ul>	<ul style="list-style-type: none"> <li>- CNN</li> </ul>	<ul style="list-style-type: none"> <li>- P= 99,98%</li> </ul>	<p>Faible</p>

<p>[37] 2020</p>	<p>Véhicule</p>	<ul style="list-style-type: none"> <li>- Données de capteurs (caméras, lidar, radar, capteurs de vitesse)</li> </ul>	<ul style="list-style-type: none"> <li>- Le freinage brusque</li> <li>- L'accélération</li> <li>- Les changements de voie brusques</li> <li>- La vitesse excessive ou insuffisante</li> </ul>	<ul style="list-style-type: none"> <li>- CNN</li> <li>- Processus gaussien (PG)</li> </ul>		<p>Moyen</p>
<p>[41] 2021</p>	<p>Véhicule</p>	<ul style="list-style-type: none"> <li>- Les indicateurs de vitesse</li> <li>- L'accélération</li> <li>- La position latérale</li> <li>- La distance</li> <li>- Le temps de collision entre les véhicules</li> </ul>	<ul style="list-style-type: none"> <li>- La présence ou l'absence d'interactions de conduite atypiques</li> </ul>	<ul style="list-style-type: none"> <li>- Auto-encodeur</li> </ul>		<p>Moyen</p>
<p>[24] 2021</p>	<p>Véhicule</p>	<ul style="list-style-type: none"> <li>- Vitesse</li> <li>- Accélérateur</li> <li>- Pression des freins</li> <li>- Angle de direction</li> </ul>	<ul style="list-style-type: none"> <li>- Comportements de conduite sûrs ou dangereux</li> </ul>	<ul style="list-style-type: none"> <li>- SVM</li> <li>- Réseau neuronal à propagation avant</li> </ul>	<ul style="list-style-type: none"> <li>- P(NN)= 90%</li> <li>- P(SVM)= 89%</li> </ul>	<p>Moyen</p>
<p>[27] 2022</p>	<p>Véhicule</p>	<ul style="list-style-type: none"> <li>- La vitesse, l'accélération</li> <li>- L'angle de virage</li> <li>- La position GPS</li> <li>- Les données météorologiques</li> </ul>	<ul style="list-style-type: none"> <li>- Les comportements de conduite sont dangereux ou non dangereux</li> </ul>	<ul style="list-style-type: none"> <li>- SAX-DF et l'apprentissage profond</li> </ul>	<ul style="list-style-type: none"> <li>- R= 89.25%</li> <li>- F1-Score= 83.62%</li> </ul>	<p>Elevé</p>
<p>[49] 2022</p>	<p>Véhicule</p>	<ul style="list-style-type: none"> <li>- Données de trajectoires de conduite</li> </ul>	<ul style="list-style-type: none"> <li>- Détection des comportements anormaux de conduite</li> </ul>	<ul style="list-style-type: none"> <li>- Variation Auto-encoder (VAE)</li> </ul>	<ul style="list-style-type: none"> <li>- Précision</li> <li>- Rappel</li> <li>- F1-score</li> </ul>	<p>Moyen</p>

<p>[26] 2023</p>	<p>Véhicule</p>	<ul style="list-style-type: none"> <li>- La vitesse</li> <li>- L'angle de tête</li> <li>- Les données contextuelles (météorologiques)</li> </ul>	<ul style="list-style-type: none"> <li>- Détection des comportements de conduite dangereux</li> </ul>	<ul style="list-style-type: none"> <li>- SAX</li> <li>- LSTM</li> </ul>	<ul style="list-style-type: none"> <li>- P= 85%</li> <li>- R= 78%</li> <li>- F1-score = 83%</li> </ul>	<p>Très Elevé</p>
<p>[30] 2016</p>	<p>Hybride</p>	<ul style="list-style-type: none"> <li>- Temps de réaction des conducteurs</li> <li>- Vitesses</li> <li>- accélérations des véhicules</li> <li>- Paramètres de communication DSRC</li> </ul>	<ul style="list-style-type: none"> <li>- La Détection des collisions</li> <li>- Les avertissements de collision erronés</li> </ul>	<ul style="list-style-type: none"> <li>-Régression logistique</li> </ul>	<ul style="list-style-type: none"> <li>- P = 67%</li> <li>- FPR = 4%</li> </ul>	<p>Moyen</p>
<p>[43] 2021</p>	<p>Hybride</p>	<ul style="list-style-type: none"> <li>- Comportements de Conduite Dangereux</li> <li>- Données Environnementales</li> </ul>	<ul style="list-style-type: none"> <li>- Risques d'accidents</li> <li>- Fréquence d'accidents</li> </ul>	<ul style="list-style-type: none"> <li>- AD</li> <li>- KNN</li> <li>- régression linéaire</li> <li>- CNN</li> <li>- AE débruteurs empilés et convolutionnels empilés</li> <li>- LSTM</li> </ul>	<ul style="list-style-type: none"> <li>- MAE=54%</li> <li>- RMSE= 18%</li> </ul>	<p>Très Elevé</p>
<p>[21] 2022</p>	<p>Hybride</p>	<ul style="list-style-type: none"> <li>- Données de trajectoires de véhicule</li> <li>- Accéléromètre</li> <li>- Gyroscope</li> </ul>	<ul style="list-style-type: none"> <li>- L'accélération</li> <li>- La décélération anormale</li> <li>- Les changements de voie</li> </ul>	<ul style="list-style-type: none"> <li>- RF</li> <li>- InceptionV3 avec Mixup</li> <li>- Smote</li> <li>- LSTM</li> </ul>	<ul style="list-style-type: none"> <li>- P(RF)= 92,25%</li> <li>- R(RF)= 91,76%</li> <li>-F1-Score (RF) =91%</li> </ul>	<p>Très Elevé</p>

TABLE 2.1 – Tableau comparatif.

### 2.4.3 Discussion

Après avoir examiné plusieurs études existantes dans la littérature sur les technologies de détection des comportements dangereux au volant, qui utilisent diverses méthodes pour repérer les comportements de conduite anormaux et renforcer la sécurité en se basant sur des caractéristiques comportementales, il est regrettable de constater que ces approches présentent plusieurs inconvénients qui nuisent à la qualité de la détection des comportements dangereux. Parmi les principaux défis, on trouve souvent des ensembles de données incomplets ou de qualité variable, ce qui peut compromettre la précision des modèles de détection. En effet, la disponibilité et la qualité des données peuvent varier en fonction des sources de données utilisées, des capteurs du véhicule, des conditions de conduite et des conditions météorologiques. Cette diversité des données peut rendre difficile la généralisation des modèles de détection à différentes situations et environnements routiers, limitant ainsi leur efficacité dans la prévention des comportements dangereux. De plus, le coût associé à la collecte, au traitement et à l'annotation des données peut également constituer une contrainte importante, notamment dans le cas de grands ensembles de données. Par conséquent, il est impératif de développer des approches capables de traiter efficacement tous types de données afin d'améliorer la fiabilité des systèmes de détection des comportements dangereux sur la route.

Les comportements des conducteurs, tels que l'utilisation du téléphone, la fatigue, l'imprudence, etc., exercent une influence considérable sur leur façon de conduire. Pour traiter ces comportements, diverses méthodes ont été utilisées, notamment la modélisation du conducteur personnalisée [18] qui se concentre uniquement sur la conduite longitudinale et ne prend pas en compte la conduite latérale (limitation de donnée de test de véhicules), rendant ainsi difficile sa généralisation. D'autres approches ont utilisé des réseaux neuronaux convolutifs CNN comme dans [52][39] il se concentre juste sur visage et main droite de conducteur, ainsi que l'approche du classifieur naïf de Bayes dans [45]. Bien que ces méthodes permettent un traitement efficace des images et des vidéos capturées, elles ne suffisent pas à détecter les comportements dangereux, car elles ne prennent pas en compte des données de conduite essentielles et plus significatives telles que la vitesse, la pression de freinage, l'orientation et la direction du véhicule, ainsi que les conditions environnementales (pluie, visibilité, etc.).

D'autres travaux ont exploité les données de conduite capturées par des accéléromètres et des gyroscopes, ainsi que par GPS, en utilisant différents classificateurs tels que SVM dans [12][24][51], et d'autres comme réseaux de neurones dans [37][38][41][48][49] comme limite l'incertitude de détection (y a d'autre situation anormale que ne sont pas découvertes), par exemple dans [41] se focaliser spécifiquement sur l'analyse des comportements de conduite longitudinale. Aussi, ces études n'ont pas pris en compte les données météorologiques, bien qu'elles puissent présenter un danger sur les routes (par exemple, une chaussée glissante due à la pluie). Un autre concept étudié est la fusion de données, abordée dans [26][27]. Cette approche combine deux types de données, à savoir les données du véhicule et celles de l'environnement. Bien que cette méthode intègre toutes les conditions nécessaires pour détecter les comportements dangereux de conduite, elle est également très coûteuse à mettre en œuvre, ce qui pourrait limiter sa faisabilité pratique.

Les approches hybrides qui ont travaillé à la fois avec les données du véhicule et celles du conducteur comme dans [30][21]. Cependant, l'absence d'utilisation des données météorologiques peut rendre ces approches moins pratiques. L'étude présentée dans [43], qui prend en compte toutes les conditions nécessaires, mais elle est coûteuse, offre une perspective difficilement réalisable en pratique.

## 2.5 Conclusion

Ce deuxième chapitre nous a permis d'explorer les différents systèmes utilisés pour la détection des comportements dangereux de conduite dans le but d'améliorer la sécurité routière. Après avoir comparé les trois approches les plus utilisées, à savoir l'approche centrée sur le conducteur, celle centrée sur le véhicule, et l'approche hybride combinant les deux, nous avons constaté que les performances dépendent de plusieurs facteurs qui peuvent varier d'un système à l'autre.

Dans le chapitre suivant, nous présenterons une nouvelle approche qui s'appuie sur l'intégration d'une méthode étudiée dans le type d'apprentissage supervisé afin de bénéficier des avantages de chaque type.

# 3

## Approche proposée (LSTM-RF)

### 3.1 Introduction

L'identification des différents comportements de conduite des conducteurs, qu'il s'agisse de conduite normale ou anormale est importante pour la sécurité routière. Détecter les comportements de conduite dangereux permet non seulement de prévenir les accidents, mais également d'améliorer la gestion du trafic. Ce chapitre détaille notre approche pour la reconnaissance des comportements de conduite en utilisant des techniques d'apprentissage automatique. Nous commencerons par discuter de la motivation liée à l'identification de ces comportements. Ensuite, nous présenterons notre méthodologie en décrivant chaque étape avec précision : de la collecte et du prétraitement des données à l'implémentation des modèles utilisés.

### 3.2 Motivation

En détectant précocement les comportements dangereux, il est possible de prévenir les accidents et d'améliorer la gestion du trafic. Cela contribue non seulement à la protection des vies humaines, mais aussi à l'optimisation des flux de circulation, réduisant ainsi les embouteillages et les coûts économiques associés aux accidents.

Comme abordé dans le chapitre précédent, il existe quelques approche qui se concentrent uniquement sur les manoeuvres longitudinales [18] [41] telles que l'accélération, le freinage et le dépassement de vitesse, et ne prend pas en compte la conduite latérale, comme le changement de voie et le maintien de la position dans la voie, en raison de la limitation des données de test des véhicules, rendant ainsi difficile sa généralisation. Cela peut réduire son efficacité dans des situations complexes, telles que la conduite en milieu urbain dense, où les véhicules doivent fréquemment changer de voie pour éviter les obstacles, réaliser des dépassements sur des routes étroites, gérer les intersections et réagir rapidement aux mouvements imprévisibles des autres usagers de la route.

Notre proposition vise à surmonter ces limites en utilisant l'apprentissage automatique. En exploitant les données de conduite collectées à partir de capteurs, il devient envisageable de développer des systèmes intelligents capables de reconnaître en temps réel les comportements de conduite à risque.

Les modèles de réseaux de neurones récurrents, notamment les LSTM, sont particulièrement adaptés à cette tâche en raison de leur capacité à traiter des séquences temporelles de données. Leur utilisation permet de capturer les dynamiques complexes des comportements de conduite sur des périodes prolongées. Parallèlement, les RF offrent une approche pour la classification grâce à leur capacité à gérer des ensembles de données diversifiés et à fournir des prédictions précises. En combinant les modèles LSTM et Random Forest (LSTM-RF), nous exploitons les avantages des deux méthodes pour améliorer la reconnaissance des comportements de conduite.

L'enjeu de ce travail est double : démontrer l'efficacité des modèles LSTM, et sa combinaison avec RF (LSTM-RF) dans la reconnaissance des comportements de conduite, et proposer une méthodologie complète et reproductible pour l'implémentation de ces techniques. En abordant ces aspects, nous visons à contribuer à la recherche en sécurité routière et à offrir des solutions concrètes pour réduire les accidents de la route.

### 3.3 Notre Proposition

Pour aborder la problématique de la détection des comportements de conduite dangereux, nous avons développé une solution basée sur l'utilisation de techniques avancées d'apprentissage automatique. Cette solution s'articule autour de deux principaux modèles : les réseaux de neurones récurrents LSTM et les RF. Cette approche combine les forces de ces deux méthodes pour offrir une reconnaissance précise et fiable des comportements de conduite. La figure 3.1 montre la structure de notre solution proposée.

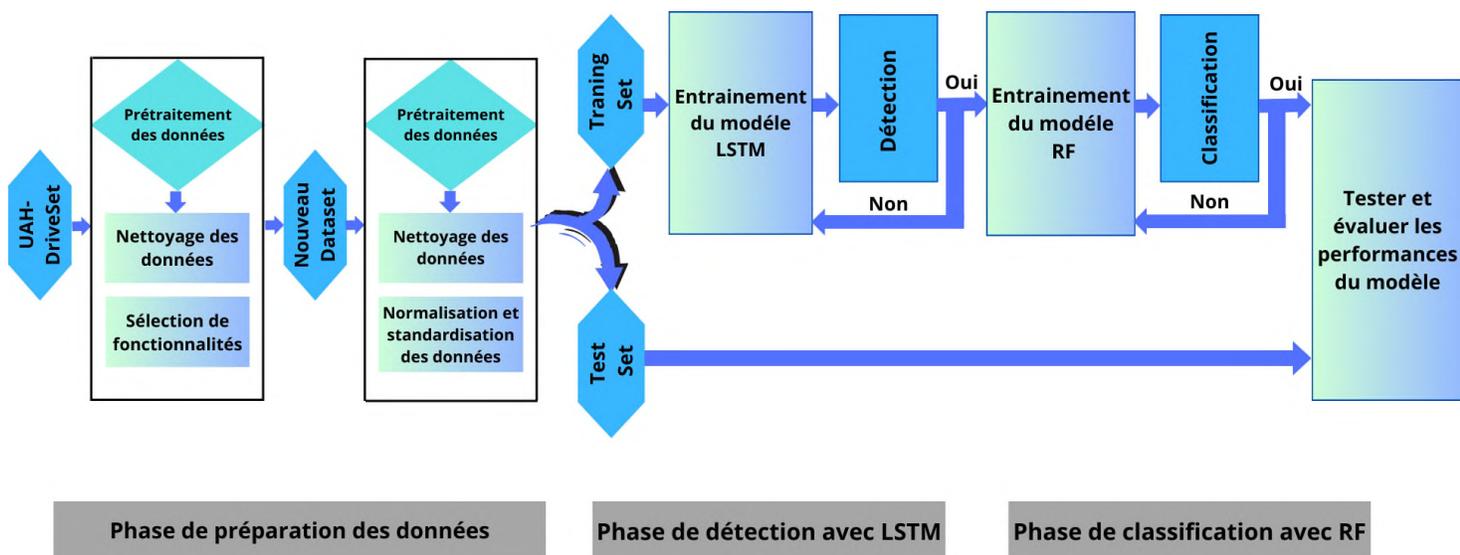


FIGURE 3.1 – Structure de la solution proposée.

### 3.4 UAH-DriveSet

Le UAH-DriveSet [35] est un ensemble de données accessible au public qui permet une analyse approfondie de la conduite. Il a été enregistré à l'aide de l'application smartphone DriveSafe, qui utilise tous les capteurs disponibles sur le smartphone (capteurs inertiels, GPS, caméra et accès Internet) pour enregistrer et reconnaître les manœuvres de conduite et en déduire les comportements associés. L'ensemble

de données comprend plus de 500 minutes de conduite naturaliste (NDS), avec des données brutes et des données traitées associées, ainsi que des enregistrements vidéo des trajets (voir le tableau 3.1). Le UAH-DriveSet permet de comprendre différentes manœuvres de conduite telles que l'accélération, le freinage, le virage, le changement de voie, la dérive de voie, le dépassement de vitesse et le suivi de voiture. De plus, il permet d'estimer le style de conduite, qu'il soit normal, somnolent ou agressif.

Catégorie	Attribut	Description	Unité de mesure	Sous-attributs
Données brutes	RAW GPS	Données collectées par le GPS	— Secondes — km/h — degrés	Horodatage, Vitesse, Latitude, Longitude, Altitude, Précision verticale, Précision horizontale, Cap, Variation de cap
	RAW ACCELEROMETERS	Mesures des capteurs inertiels (accéléromètres)	— Secondes — Gs — degrés	Horodatage, Booléen du système activé, Accélération en X, Accélération en Y, Accélération en Z, Accélération en X filtrée par KF, Accélération en Y filtrée par KF, Accélération en Z filtrée par KF, Roulis, Tangage, Lacet
Données traitées	PROC LANE DETECTION	Variables basées sur la vision et la détection de voie	— Secondes — mètres — degrés	Horodatage, Position de la voiture par rapport au centre de la voie, Angle de la voiture par rapport à la courbure de la voie, Largeur de la route, État de l'algorithme de détection de voie
	PROC VEHICLE DETECTION	Informations sur les véhicules détectés	— Secondes — mètres — km/h	Horodatage, Distance au véhicule en amont dans la voie actuelle, Temps d'impact avec le véhicule en amont, Nombre de véhicules détectés dans cette image, Vitesse GPS

TABLE 3.1 – Attributs du UAH-DriveSet.

### 3.5 Phase de préparation des données

Dans le cadre de notre simulation, nous n'allons pas utiliser tous les attributs du jeu de données. En effet, pour optimiser la performance de notre modèle et éviter la surcharge d'informations, nous avons sélectionné un sous-ensemble d'attributs pertinents qui sont les plus susceptibles d'influencer le comportement de conduite. Ces attributs ont été choisis en fonction de leur pertinence et de leur importance dans la prédiction des manœuvres de conduite. La figure 3.2 illustre les attributs que nous avons décidé d'utiliser pour notre simulation.

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4799 entries, 0 to 4798
Data columns (total 11 columns):
#   Column                                                                 Non-Null Count  Dtype
---  -
0   Timestamp                                                              4799 non-null  float64
1   Speed                                                                  4799 non-null  float64
2   X_car_position_relative_lane_center 4799 non-null  float64
3   W_road_width                                                           4799 non-null  float64
4   State_lane_det                                                         4799 non-null  int64
5   Distance_vehicle_current_lane    4799 non-null  float64
6   Time_impact_ahead_vehicle        4799 non-null  float64
7   Number_detected_vehicles_frame   4799 non-null  int64
8   Maximum_allowed_speed_current_road 4799 non-null  float64
9   Number_lanes_current_road        4799 non-null  int64
10  Estimated_current_lane           4799 non-null  int64
dtypes: float64(7), int64(4)
memory usage: 412.5 KB

```

FIGURE 3.2 – Attributs du dataset.

### 3.5.1 Prétraitement des données

Le prétraitement des données de conduite est une phase fondamentale dans le traitement des données associées à la conduite. Son objectif est de préparer les données non traitées pour l'analyse et la modélisation. Cette phase englobe plusieurs étapes, telles que la collecte des données, l'épuration des données, la normalisation des données, la transformation des données et l'extraction des caractéristiques.

Voici un aperçu détaillé des étapes du prétraitement des données de conduite :

1. **Nettoyage des données** : Cette phase implique l'élimination des erreurs et des données aberrantes, le remplacement des valeurs manquantes et la suppression des bruits. Les erreurs dans les données peuvent être dues à des capteurs défectueux ou à des perturbations environnementales, tandis que les données aberrantes peuvent résulter d'événements inattendus tels que des accidents ou des incidents de conduite. Il est essentiel de remplacer les valeurs manquantes pour éviter les erreurs dans les analyses et les modèles, et l'élimination des bruits est importante pour minimiser la variabilité dans les données. Les techniques fréquemment utilisées pour le nettoyage des données comprennent la détection de données aberrantes, l'imputation de données manquantes et le filtrage des données.
2. **Sélection de fonctionnalités** : Nous avons enrichi le jeu de données initial en ajoutant des caractéristiques pertinentes pour notre analyse. Ces caractéristiques ont été calculées en fonction de différents aspects du comportement de conduite des véhicules enregistrés. Voici un aperçu des caractéristiques ajoutées :

- **Excès de vitesse**

Nous avons commencé par calculer le ratio de la vitesse actuelle du véhicule par rapport à la vitesse maximale autorisée sur la route concernée. Ce ratio, appelé "ratio\_exces\_vitesse", permet d'évaluer si le véhicule dépasse la limite de vitesse. La formule utilisée est la suivante :

$$\text{ratio\_exces\_vitesse} = \frac{\text{Speed}}{\text{Maximum\_allowed\_speed\_current\_road}} \quad (3.1)$$

Ensuite, une colonne supplémentaire, "exces\_vitesse", a été ajoutée pour signaler les cas où ce ratio dépasse 1, indiquant ainsi un excès de vitesse. Formellement :

$$\text{exces\_vitesse} = \begin{cases} 1 & \text{si ratio\_exces\_vitesse} > 1 \\ 0 & \text{sinon} \end{cases} \quad (3.2)$$

- **Proximité de distance**

Nous avons élaboré une fonction pour calculer la distance de sécurité entre les véhicules, en prenant en compte la vitesse du véhicule et le temps de réaction du conducteur. Cette distance de sécurité, nommée "distance\_securitee", est calculée selon la formule [36] [15] :

$$\text{distance\_securitee} = \left( \frac{\text{vitesse}}{3.6} \right) \times \text{temps\_reaction} \quad (3.3)$$

Où

- La vitesse est convertie de km/h en m/s en divisant par 3.6,
- Le temps de réaction qui varie de 1 à 2 secondes, est pris comme 1,5 seconde.

Ensuite a été utilisée pour déterminer si la distance entre les véhicules était inférieure à la distance sécurisée. Une colonne supplémentaire nommée "proximité\_dangereuse" a été ajoutée pour signaler de telles situations de proximité dangereuse. Formellement :

$$\text{proximité\_dangereuse} = \begin{cases} 1 & \text{si Distance\_vehicule\_current\_lane} < \text{distance\_securitee} \\ 0 & \text{sinon} \end{cases} \quad (3.4)$$

- **Exemple :** Prenons un véhicule roulant à 90 km/h. La distance de sécurité, calculée avec une conversion de la vitesse en m/s (90 km/h équivaut à 25 m/s) et un temps de réaction de 1,5 seconde, est :

$$\text{distance\_securitee} = 25 \times 1.5 = 37.5$$

Ainsi, pour un véhicule à 90 km/h, la distance de sécurité recommandée est de 37,5 mètres. Comme illustré dans la figure 3.3, la distance entre les véhicules est comparée à la distance de sécurité pour déterminer si la situation est dangereuse ou non.

- Flèche rouge : Indique une distance de 25 mètres, ce qui est inférieur à la distance de sécurité de 37,5 mètres, signalant une situation dangereuse.
- Flèche verte : Indique une distance de 50 mètres, ce qui est supérieur à la distance de sécurité, signalant une situation sûre.

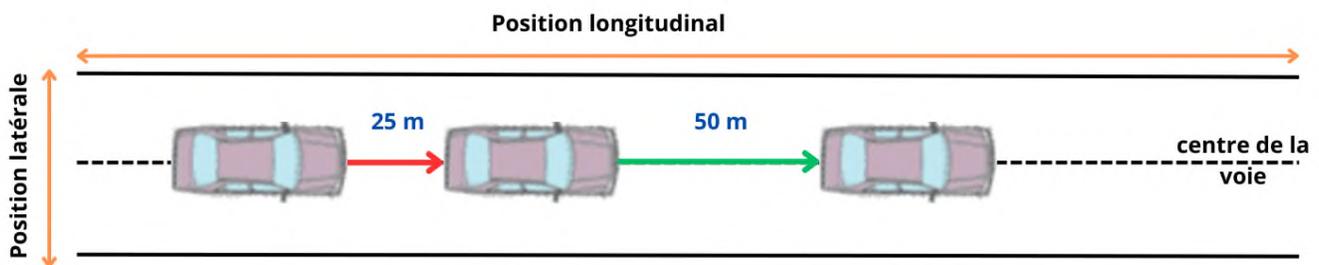


FIGURE 3.3 – Illustration de la distance de sécurité entre les véhicules.

- **Conduite sur voie étroite**

Pour détecter la conduite sur une voie étroite, nous avons défini des seuils basés sur la largeur de la route et la variation de la position latérale du véhicule. La variation de la position latérale ( $\Delta x$ ) est calculée par :

$$\Delta x = \frac{\Delta X_{\text{car\_position\_relative\_lane\_center}}}{\Delta \text{Timestamp}} \quad (3.5)$$

Si la largeur de la route était inférieure à un seuil prédéfini ( $\text{road\_width\_threshold} = 3.0$ ) et que la variation de la position latérale était significative ( $\Delta x_{\text{threshold}} = 0.5$ ), alors la colonne "conduite\_voie\_etroite" était marquée comme étant à 1 pour indiquer une conduite sur une voie étroite. Formellement :

$$\text{conduite\_voie\_etroite} = \begin{cases} 1 & \text{si } (W_{\text{road\_width}} < \text{road\_width\_threshold}) \\ & \wedge (|\Delta x| > \Delta x_{\text{threshold}}) \\ 0 & \text{sinon} \end{cases} \quad (3.6)$$

- **Exemple :** Supposons un véhicule sur une route de largeur 2.5 mètres (ce qui est inférieur au seuil de 3.0 mètres) et sa position latérale change souvent.

- À  $t_1$  le véhicule est à 0.2 m du centre de la voie.

- À  $t_2$  (après 1 seconde) le véhicule est à 1.0 m du centre de la voie.
- La variation de la position latérale est :

$$\Delta x = \frac{1.0 - 0.2}{1} = 0.8 \text{ m/s}$$

Ce résultat est supérieur au seuil de 0.5 m/s. Donc, selon les critères définis, la conduite sur une voie étroite est détectée.

- **Détection des zigzags**

Les mouvements en zigzag sont des manoeuvres en forme de "Z" répétées d'un véhicule dans une voie. Pour identifier ces mouvements, nous calculons le rapport entre la position latérale du véhicule et la largeur de la voie. Ce rapport (rapport\_x\_w) est défini par :

$$\text{rapport\_x\_w} = \frac{X_{\text{car\_position\_relative\_lane\_center}}}{W_{\text{road\_width}}} \quad (3.7)$$

Nous identifions les variations importantes de ce rapport pour détecter les zigzags. La variation du rapport (delta\_rapport\_x\_w) est définie par :

$$\Delta \text{rapport\_x\_w} = \text{rapport\_x\_w}(t) - \text{rapport\_x\_w}(t - 1) \quad (3.8)$$

Un zigzag est détecté si cette variation dépasse un seuil prédéfini (zigzag\_threshold = 0.1). Formellement :

$$\text{zigzag} = \begin{cases} 1 & \text{si } |\Delta \text{rapport\_x\_w}| > \text{zigzag\_threshold} \\ 0 & \text{sinon} \end{cases} \quad (3.9)$$

- **Exemple**

- La largeur de la route  $W$  est 3.2 mètres. Les positions latérales du véhicule à différents instants sont :
  - $t_1$  :  $X(1) = 0$  m,
  - $t_2$  :  $X(2) = 1.5$  m,
  - $t_3$  :  $X(3) = 0.8$  m,
  - $t_4$  :  $X(4) = 0.4$  m.
- Pour chaque instant  $t$ , calculons le rapport rapport\_x\_w ( $t$ ) :

$$\begin{aligned} \text{rapport\_x\_w (1)} &= \frac{0}{3.2} = 0, \\ \text{rapport\_x\_w (2)} &= \frac{1.5}{3.2} \approx 0.469, \\ \text{rapport\_x\_w (3)} &= \frac{0.8}{3.2} \approx 0.250, \\ \text{rapport\_x\_w (4)} &= \frac{0.4}{3.2} \approx 0.125. \end{aligned}$$

— Pour chaque instant  $t$  (à partir de  $t_2$ ), calculons la variation  $\Delta \text{rapport\_x\_w}(t)$  :

$$\begin{aligned} \Delta \text{rapport\_x\_w (2)} &= 0.469 - 0 = 0.469, \\ \Delta \text{rapport\_x\_w (3)} &= 0.250 - 0.469 = -0.219, \\ \Delta \text{rapport\_x\_w (4)} &= 0.125 - 0.250 = -0.125. \end{aligned}$$

— En utilisant le seuil prédéfini  $\text{zigzag\_threshold} = 0.1$ , vérifions si  $|\Delta \text{rapport\_x\_w}(t)|$  dépasse ce seuil pour chaque instant :

$$\begin{aligned} \text{Pour } t_2 : |\Delta \text{rapport\_x\_w (2)}| &> 0.1 \rightarrow \text{zigzag} = 1, \\ \text{Pour } t_3 : |\Delta \text{rapport\_x\_w (3)}| &> 0.1 \rightarrow \text{zigzag} = 1, \\ \text{Pour } t_4 : |\Delta \text{rapport\_x\_w (4)}| &> 0.1 \rightarrow \text{zigzag} = 1. \end{aligned}$$

La figure 3.4 représente l'exemple de détection des zigzags :

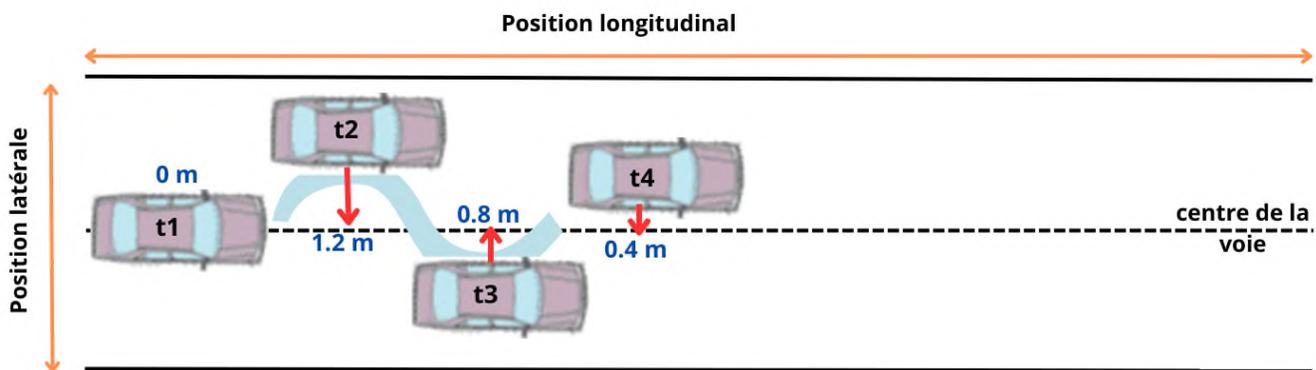


FIGURE 3.4 – Illustration de la détection des zigzags.

- **Classe normal/anormal**

Nous avons créé deux nouvelles colonnes, 'normal' et 'anormal', que nous avons ensuite mises à jour en fonction des conditions données. Si une observation ne présentait aucun excès de vitesse, zigzag, proximité dangereuse ou conduite sur une voie étroite, elle était classée comme 'normale'. En revanche, si une de ces conditions était remplie, l'observation était

classée comme 'anormale'. Formellement, pour une observation  $i$  :

$$\text{normal}_i = \begin{cases} 1 & \text{si } \text{exces\_vitesse}_i = 0 \wedge \text{zigzag}_i = 0 \\ & \wedge \text{proximite\_dangereuse}_i = 0 \wedge \text{conduite\_voie\_etroite}_i = 0 \\ 0 & \text{sinon} \end{cases} \quad (3.10)$$

$$\text{anormal}_i = \begin{cases} 1 & \text{si } \text{exces\_vitesse}_i = 1 \vee \text{zigzag}_i = 1 \\ & \vee \text{proximite\_dangereuse}_i = 1 \vee \text{conduite\_voie\_etroite}_i = 1 \\ 0 & \text{sinon} \end{cases} \quad (3.11)$$

Nous avons créé un nouveau jeu de données en utilisant les attributs de l'ensemble de données original (UAH-DriveSet). La figure 3.5 montre le dataset final sur lequel nous avons entraîné le modèle.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4799 entries, 0 to 4798
Data columns (total 24 columns):
#   Column                                     Non-Null Count  Dtype
---  -
0   Timestamp                                 4799 non-null   float64
1   Speed                                     4799 non-null   float64
2   X_car_position_relative_lane_center      4799 non-null   float64
3   W_road_width                             4799 non-null   float64
4   State_lane_det                           4799 non-null   int64
5   Distance_vehicle_current_lane           4799 non-null   float64
6   Time_impact_ahead_vehicle               4799 non-null   float64
7   Number_detected_vehicles_frame          4799 non-null   int64
8   Maximum_allowed_speed_current_road      4799 non-null   float64
9   Number_lanes_current_road               4799 non-null   int64
10  Estimated_current_lane                   4799 non-null   int64
11  ratio_exces_vitesse                      4799 non-null   float64
12  exces_vitesse                            4799 non-null   int64
13  distance_securitee                       4799 non-null   float64
14  distance_proximite_dangereuse            4799 non-null   bool
15  temps_impact_proximite_dangereuse        4799 non-null   bool
16  proximite_dangereuse                     4799 non-null   int64
17  delta_x                                  4799 non-null   float64
18  conduite_voie_etroite                   4799 non-null   int64
19  rapport_x_w                              4799 non-null   float64
20  delta_rapport_x_w                        4799 non-null   float64
21  zigzag                                   4799 non-null   int64
22  normal                                   4799 non-null   int64
23  anormal                                  4799 non-null   int64
dtypes: bool(2), float64(12), int64(10)
memory usage: 834.3 KB
```

FIGURE 3.5 – Attributs du nouveau dataset.

3. **Normalisation et standardisation des données** : Cette phase implique de mettre les données sur une échelle uniforme pour faciliter leur manipulation. Les données recueillies peuvent être dans des formats variés et sur des échelles différentes, ce qui peut compliquer leur comparaison et leur intégration dans les modèles. La normalisation des données consiste à transformer les données pour les aligner sur une échelle commune, ce qui facilite la comparaison des données et leur utilisation dans les modèles. Les techniques fréquemment utilisées pour la normalisation des données comprennent la normalisation min-max et la normalisation z-score. La formule mathématique pour la normalisation Min-Max est la suivante [2].

$$X_{\text{norm}} = \frac{X - X_{\text{min}}}{X_{\text{max}} - X_{\text{min}}} \quad (3.12)$$

Où :

- $X_{\text{norm}}$  est la valeur normalisée,
- $X$  est la valeur originale,
- $X_{\text{min}}$  est la valeur minimale dans l'ensemble de données,
- $X_{\text{max}}$  est la valeur maximale dans l'ensemble de données.

Ainsi la formule mathématique pour la normalisation z-score [2] :

$$Z = \frac{X - \mu}{\sigma} \quad (3.13)$$

Où :

- $Z$  est la valeur normalisée (Z-score),
- $X$  est la valeur originale,
- $\mu$  est la moyenne de l'ensemble de données,
- $\sigma$  est l'écart type de l'ensemble de données.

### 3.6 Phase de détection avec LSTM

Pour cette phase, nous avons utilisé un modèle LSTM pour détecter les comportements de conduite en fonction des caractéristiques extraites des données, la figure 3.6 illustre notre modèle LSTM :

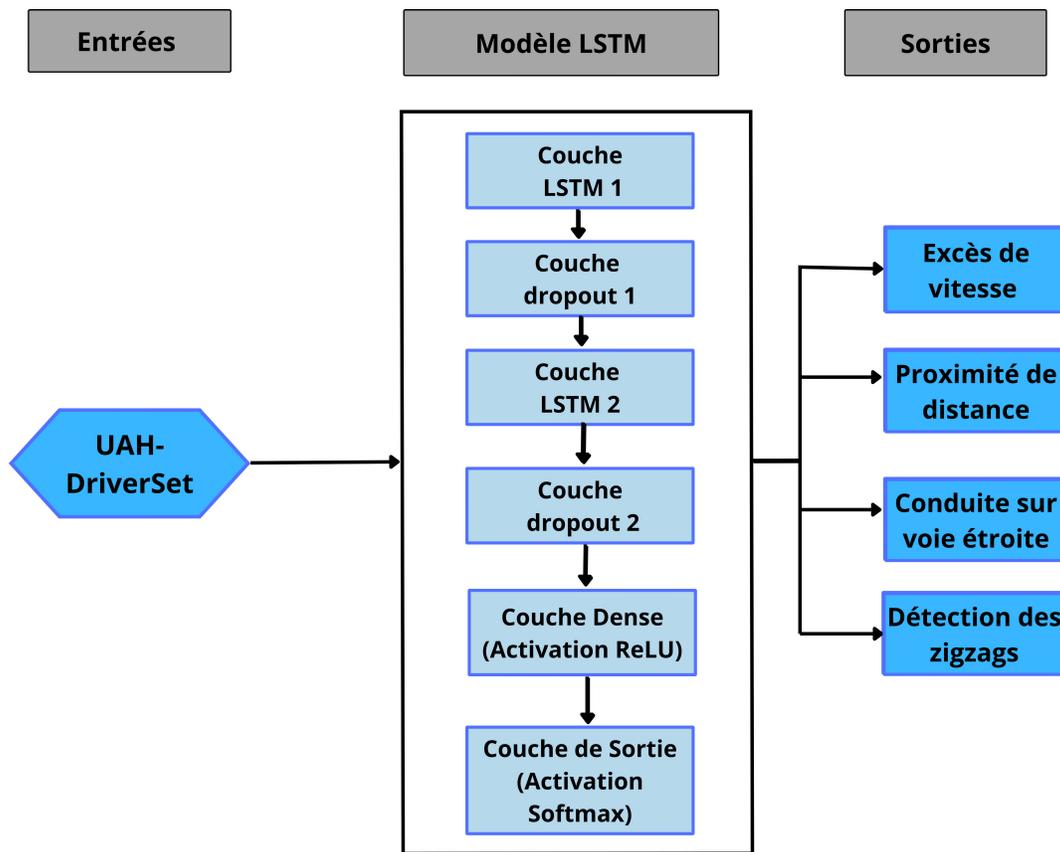


FIGURE 3.6 – Diagramme de détection par LSTM.

- **Les couches LSTM**

Nous commençons par la première couche avec 50 unités et `'return_sequences=True'`, Cette configuration permet à la couche de renvoyer la séquence complète de sorties pour chaque pas de temps. Cette couche capture les dépendances temporelles à partir des séquences d'entrée.

Ensuite, une deuxième couche est intégrée, également avec 50 unités mais avec `'return_sequences=False'`. Cette couche traite les sorties de la première couche et ne renvoie que la dernière sortie de la séquence, réduisant ainsi la dimensionnalité et facilitant la prise de décision finale.

- **Les couches Dropout**

Nous utilisons deux couches Dropout pour régulariser le modèle et prévenir le surapprentissage. La première est ajoutée après la première couche LSTM avec un taux de 0.2, mettant aléatoirement à zéro 20% des unités pour éviter que les neurones ne deviennent trop dépendants de certaines caractéristiques. La seconde, également avec un même taux, est placée après la deuxième couche LSTM pour fournir une régularisation supplémentaire et réduire davantage le risque de surapprentissage.

- **Couche dense avec activation ReLU**

Nous ajoutons une couche Dense avec 128 unités et une activation ReLU (Rectified Linear Unit). Cette couche dense aide à mapper les représentations capturées par les couches LSTM à un espace

approprié pour la détection finale. L'activation ReLU est utilisée pour introduire la non-linéarité dans le modèle, ce qui permet d'apprendre des relations plus complexes entre les caractéristiques et les cibles.

- **Couche de sortie avec activation Softmax**

Enfin, nous ajoutons une couche de sortie Dense avec une activation Softmax. Cette couche contient autant d'unités que de classes cibles (dans notre cas quatre), et l'activation Softmax est utilisée pour produire des probabilités de sortie pour chaque classe. Chaque unité de sortie représente la probabilité d'un comportement de conduite spécifique (excès de vitesse, zigzag, proximité dangereuse, conduite sur voie étroite).

### 3.7 Phase de classification avec RF

Pour améliorer la précision de notre modèle de prédiction des comportements de conduite, nous avons adopté une approche hybride en intégrant les prédictions d'un modèle LSTM comme nouvelles caractéristiques d'entrée pour le classificateur RF. Cette méthode exploite les forces des deux modèles : les capacités de séquençage et de capture de dépendances temporelles du LSTM, ainsi que l'efficacité de la Random Forest dans le traitement de données de haute dimension.

Nous avons implémenté notre modèle de RF en utilisant la classe 'RandomForestRegressor' de la bibliothèque Scikit-Learn. Un paramètre clé dans cette implémentation est 'n\_estimators', qui spécifie le nombre d'arbres dans notre forêt. Nous avons soigneusement choisi ce nombre pour trouver un équilibre optimal entre biais et variance afin de réduire le risque de surajustement tout en conservant une bonne capacité de généralisation du modèle.

La figure 3.7 illustre notre modèle RF, mettant en évidence l'intégration des prédictions LSTM dans la phase de classification de la Random Forest.

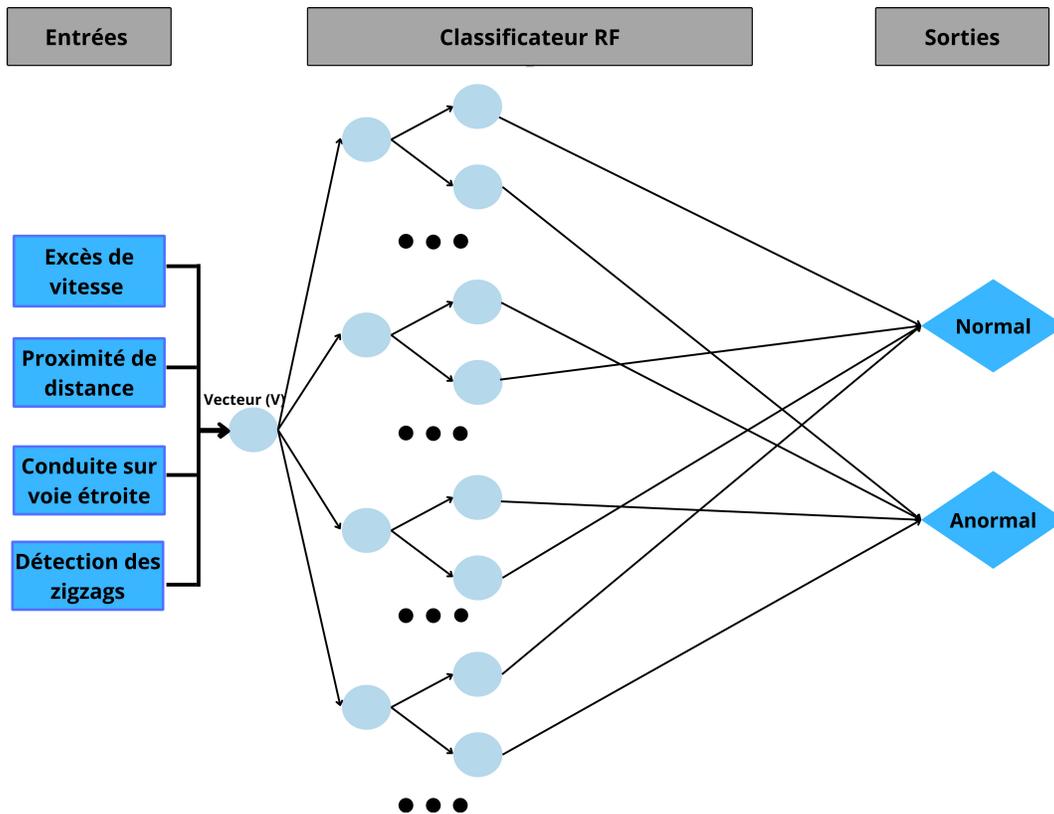


FIGURE 3.7 – Diagramme de classification par RF.

### 3.8 Conclusion

En combinant les modèles LSTM et RF, notre approche propose une méthode efficace pour la reconnaissance des comportements de conduite dangereux. Les LSTM capturent les dynamiques temporelles complexes, tandis que les Random Forest fournissent des classifications fiables. Cette combinaison permet d'améliorer la précision de la détection des comportements de conduite, contribuant ainsi à la sécurité routière et à la gestion efficace du trafic. Cette méthodologie complète et reproductible offre une base solide pour le développement de systèmes intelligents de surveillance et de prévention des accidents de la route.

# 4

## Expérimentation

### 4.1 Introduction

Dans ce chapitre, nous procéderons à une évaluation approfondie de notre modèle et analyserons les résultats obtenus à partir de simulations. Cette évaluation nous permettra d'examiner la précision, le rappel, le score  $R^2$  et le score F1 de notre modèle, ainsi que les taux d'erreurs associés. Nous commencerons par présenter les outils et bibliothèques utilisés, ainsi que les métriques de performance employées pour évaluer notre solution. Ensuite, nous fournirons une analyse détaillée des résultats obtenus à chaque phase de la simulation de la solution proposée, en incluant les comparaisons pertinentes réalisées.

### 4.2 Outils de développement

- **Google colab** : Colaboratory ou 'Colab' est une plateforme offerte gratuitement par Google, qui permet d'écrire et d'exécuter du code Python via un navigateur. Basé sur Jupyter Notebook, il est spécialement conçu pour la formation et la recherche en apprentissage automatique. Cette plateforme facilite l'entraînement de modèles de Machine Learning directement dans le cloud [14].
- **Jupyter** : Jupyter est une application web open source qui permet de créer et de partager des documents interactifs contenant du code en direct, des équations, des visualisations et des textes explicatifs. Jupyter prend en charge de nombreux langages de programmation, notamment Python.
- **Python** : Python est un langage de programmation de haut niveau, largement utilisé pour la programmation générale en raison de sa clarté et de sa facilité d'utilisation. Il dispose d'un système de typage dynamique, d'une gestion automatique de la mémoire et prend en charge plusieurs paradigmes de programmation, y compris la programmation orientée objet et fonctionnelle. Doté d'une vaste bibliothèque standard, Python est disponible sur de nombreux systèmes d'exploitation [9].

### 4.3 Bibliothèques utilisées

- **Pandas** : Est un outil d'analyse et de manipulation de données open source rapide, puissant, flexible et facile à utiliser, construit sur le langage de programmation Python [14].
- **TensorFlow** : Est une bibliothèque open source d'apprentissage automatique, créée par Google, permettant de développer et d'exécuter des applications de Machine Learning et de Deep Learning et regroupe un grand nombre de modèles et d'algorithmes de ces applications [9].
- **Scikit-learn (ou sklearn)** : Est l'une des bibliothèques les plus utiles pour l'apprentissage automatique en Python. Elle contient de nombreux outils efficaces pour l'apprentissage automatique et la modélisation statistique, notamment la classification, la régression, le clustering et la réduction de la dimensionnalité [14].
- **Keras** : Est une bibliothèque open source écrite en python et permettant d'interagir avec les algorithmes de réseaux de neurones profonds et de machine Learning, notamment Tensorflow [9].
- **Matplotlib** : Est une bibliothèque du langage de programmation Python destinée à tracer et visualiser des données sous formes de graphiques [14].
- **NumPy** : Est une bibliothèque permettant d'effectuer des calculs numériques avec Python. Elle introduit une gestion facilitée des tableaux de nombres, des fonctions sophistiquées (diffusion) [9].
- **SciPy** : Est une bibliothèque Python utilisée pour les calculs scientifiques et techniques [14].

### 4.4 Évaluation des performances du modèle

Cette section traite des mesures utilisées pour évaluer à quel point le modèle est capable de détecter et de classer correctement les comportements de conduite, ainsi que de mesurer la précision et l'erreur de prédiction du modèle.

- **Les vrais positifs (VP)** : Nombre d'activités bien prédites dans la classe à juste titre [9].
- **Les vrais négatifs (VN)** : Nombre d'activités prédites comme n'étant pas dans la classe à juste titre [9].
- **Les faux positifs (FP)** : Nombre d'activités prédites dans la classe alors qu'ils ne devraient pas en faire partie [9].
- **Les faux négatifs (FN)** : Nombre d'activités prédites comme étant de la classe alors qu'ils ne le sont pas en vrai [9].
- **Précision** : Il s'agit de la mesure de performance la plus courante pour les algorithmes de classification. Il peut être défini comme le nombre de prédictions correctes faites par rapport à toutes les prédictions faites [4]. La formule est définie comme suit [17] :

$$\text{Précision (\%)} = \left( \frac{VP}{VP + FP} \right) \times 100 \quad (4.1)$$

- **Rappel** : Le rappel (recall) est une mesure de performance utilisée en classification pour évaluer la capacité d'un modèle à identifier correctement tous les échantillons positifs parmi ceux qui sont effectivement positifs. Plus précisément, il mesure la proportion d'échantillons positifs réellement identifiés comme tels parmi tous les échantillons qui sont effectivement positifs [4]. La formule est définie comme suit [17] :

$$\text{Rappel (\%)} = \left( \frac{VP}{VP + FN} \right) \times 100 \quad (4.2)$$

- **R<sup>2</sup> Score** : Ou le coefficient de détermination (Wright, 1921) peut être interprété comme la proportion de la variance de la variable dépendante qui est prévisible à partir des variables indépendantes [13]. La valeur du R<sup>2</sup> varie de 0 à 1. Voici sa formule mathématique [33] :

$$R^2 = 1 - \frac{SSE}{SST} \quad (4.3)$$

où SSE (Sum of Squared Errors) représente la somme des carrés des écarts entre les valeurs prédites et les valeurs réelles, et SST (Total Sum of Squares) représente la somme des carrés des écarts entre les valeurs réelles et la moyenne des valeurs réelles.

- **F1-score** : Le F1-score (aussi appelé F1-mesure) est une métrique couramment utilisée pour évaluer les performances des algorithmes d'apprentissage [35]. Il représente la moyenne pondérée de la précision et du rappel. La meilleure valeur de F1-score serait 1 et la pire serait 0 [4]. La formule est donnée par [17] :

$$\text{F1-score (\%)} = 2 \times \frac{\text{Précision} \times \text{Rappel}}{\text{Précision} + \text{Rappel}} \times 100 \quad (4.4)$$

- **Erreur absolue moyenne MAE** : Le MAE (Mean Absolute Error) est une mesure d'erreur elle est simplement définie comme la moyenne de la différence absolue entre la sortie prédite et la sortie réel [16]. La formule est définie comme suit :

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i| \quad (4.5)$$

Où :

- $y_i$  : la valeur prédite,
- $\hat{y}_i$  : la valeur réelle de base,
- $N$  : le nombre de toutes les valeurs prédites [5].

- **Erreur quadratique moyenne MSE** : Le MSE (Mean Square Error) représente la moyenne de la différence au carré entre les valeurs d'origine et les valeurs prédites dans l'ensemble de données. Il mesure la variance des valeurs résiduelles. La formule est définie comme suit :

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad (4.6)$$

Où :

- $y_i$  : la valeur réelle,
  - $\hat{y}_i$  : la valeur prédite,
  - $N$  : le nombre de toutes les valeurs prédites [14].
- **Racine de l'erreur quadratique moyenne RMSE** : Le RMSE (Root Mean Squared Error) est une mesure statistique couramment utilisée pour évaluer la précision des prédictions par rapport aux valeurs observées. Il est défini comme la racine carrée de la moyenne des carrés des écarts entre les valeurs prédites et les valeurs réelles [25]. La formule est définie comme suit :

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2} \quad (4.7)$$

Où :

- $y_i$  : la valeur réelle,
  - $\hat{y}_i$  : la valeur prédite,
  - $N$  : le nombre de toutes les valeurs prédites [46].
- **La courbe ROC (Receiver Operating Characteristic)** : Est un graphique du taux de vrais positifs (TPR), également connu sous le nom de sensibilité ou rappel, par rapport au taux de faux positifs (FPR), également appelé taux de fausse alarme, d'un classificateur en faisant varier le seuil de décision. Une mesure statistique clé associée à la courbe ROC est l'aire sous la courbe (AUC), qui est largement utilisée pour évaluer la performance d'un classificateur. Une autre mesure étroitement liée est l'AUC partielle, qui correspond à l'AUC dans une région spécifique restreignant la plage de FPR et/ou TPR [47].

Le taux de vrais positifs est également appelé sensibilité telle que [11] :

$$\text{Sensitivity} = \text{TruePositives} / (\text{TruePositives} + \text{FalseNegatives}) \quad (4.8)$$

Le taux de faux positifs est également appelé spécificité inversée, où la spécificité est le nombre total de vrais négatifs divisé par la somme du nombre de vrais négatifs et de faux positifs telle que [11] :

$$\text{Specificity} = \text{TrueNegatives} / (\text{TrueNegatives} + \text{FalsePositives}) \quad (4.9)$$

## 4.5 Evaluation des performances

Dans cette section, nous aborderons l'entraînement de notre modèle, en détaillant les différentes étapes et techniques employées pour optimiser ses performances. Nous présenterons également les résultats obtenus, en les analysant de manière approfondie pour évaluer l'efficacité du modèle. Enfin, nous visualiserons les comparaisons réalisées avec d'autres modèles ou approches, afin de mieux comprendre les forces et les faiblesses de notre solution et de mettre en évidence les améliorations potentielles.

### 4.5.1 Phase de détection

Dans notre analyse, nous avons identifié les caractéristiques les plus pertinentes, telles que la vitesse du véhicule, la position latérale et la largeur de la route, pour constituer les données d'entrée de notre

modèle LSTM. Les comportements de conduite, comme l'excès de vitesse, le zigzag, la proximité dangereuse et la conduite sur voie étroite, ont été utilisés comme cibles de classification. Les données ont été divisées en ensembles d'entraînement et de test selon une répartition de 80/20 afin d'évaluer la performance du modèle. Les données d'entraînement et de test ont ensuite été réorganisées pour correspondre à la forme d'entrée requise par le LSTM. Pour l'entraînement du modèle, nous avons utilisé la fonction de perte *binary\_crossentropy*.

L'optimiseur Adam a été utilisé pour ajuster les poids du réseau en minimisant cette fonction de perte. L'entraînement du modèle s'est déroulé sur 150 époques avec un *batch size* de 42 et une validation sur 20% des données d'entraînement.

#### 4.5.1.1 Résultats Obtenus

Dans cette section, nous présentons les résultats des performances de notre modèle de détection. Les figures suivantes montrent les différentes étapes de l'entraînement du modèle et mettent en évidence les résultats obtenus en termes de précision et de perte sur les ensembles de données d'entraînement et de validation (voir la figure 4.1).

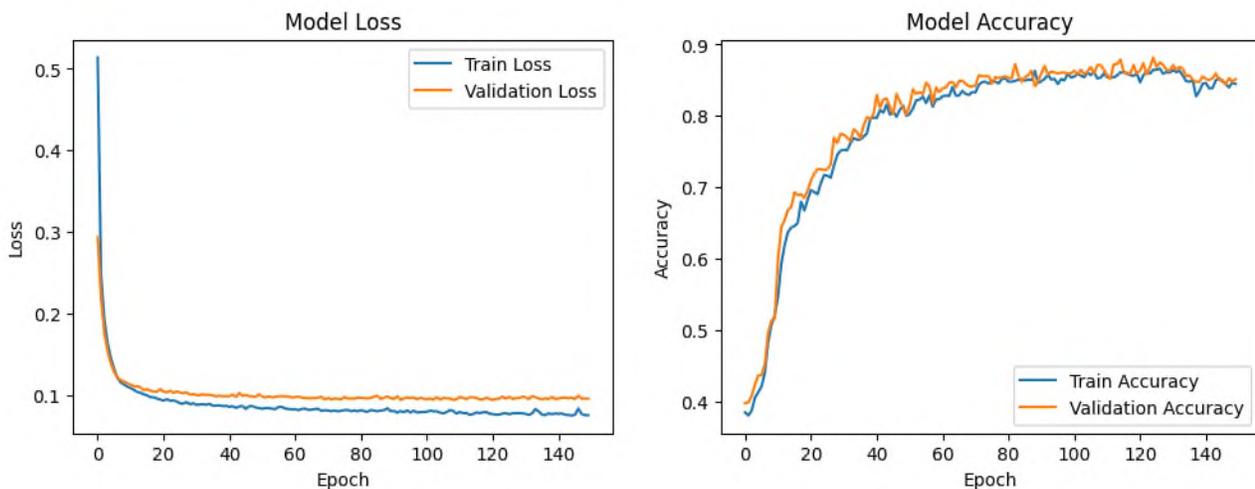


FIGURE 4.1 – Taux d'erreur et précision du LSTM.

- **Discussion :** La figure 4.1 montre l'évolution de l'exactitude et de la perte pour le modèle LSTM au cours de l'entraînement. Les courbes d'exactitude montrent une montée rapide initiale suivie d'une stabilisation autour de 85-90 %, avec les courbes d'entraînement et de validation très proches, indiquant une bonne généralisation. Les courbes de perte, quant à elles, diminuent rapidement au début puis se stabilisent avec un écart minimal entre les pertes d'entraînement et de validation, suggérant une bonne performance sans sur-apprentissage notable.

#### 4.5.1.2 Comparaison avec d'autre modèle

Pour évaluer la performance de notre modèle LSTM, nous avons testé plusieurs autres modèles, y compris les réseaux de neurones convolutifs (CNN), déjà utilisé dans le travail cité dans l'état de l'art [38], ainsi que les perceptrons multicouches (MLP) considéré comme un modèle basique des réseaux de neurones artificiels (RNA). Ce choix a été motivé par le fait que tous ces modèles appartiennent à la même approche générale de réseaux de neurones artificiels. Nous présentons des figures qui illustrent ces

différences de performance entre les différents modèles testés.

La figure 4.2 illustre les performances d'un modèle CNN au cours des 150 époques d'entraînement. Le graphe de gauche montre l'évolution de la perte pour les ensembles d'entraînement et de validation. Le graphe de droite représente la précision du modèle.

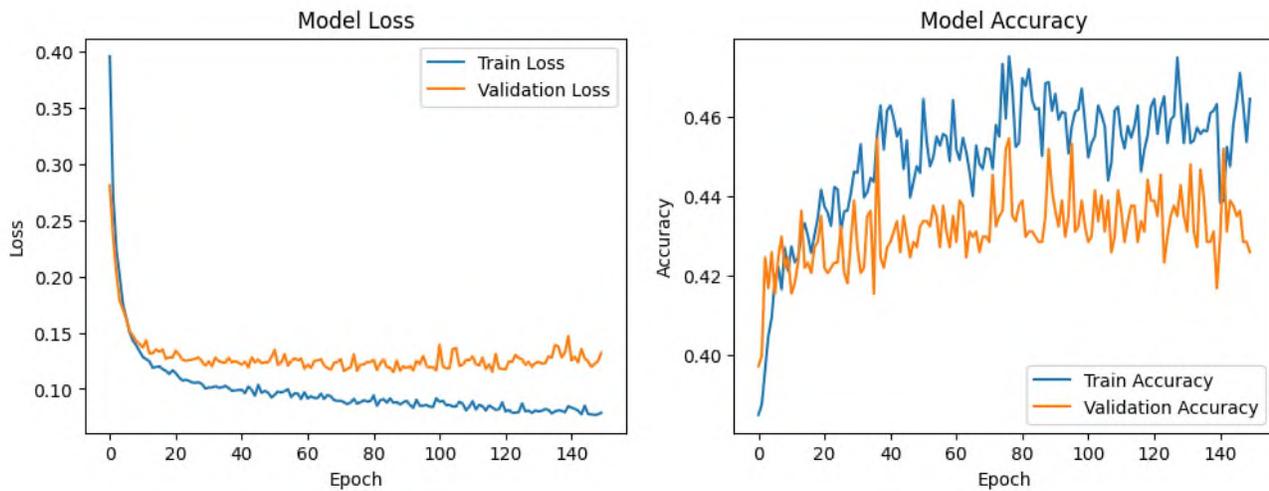


FIGURE 4.2 – Taux d’erreur et précision du CNN.

- Discussion :** La figure 4.2 montre l'évolution de la perte et de la précision pour le modèle CNN. Les courbes de perte montrent une diminution rapide initiale suivie d'une stabilisation, avec une perte d'entraînement qui continue de baisser légèrement et une perte de validation qui fluctue autour de 0.15, indiquant un possible sur-apprentissage après environ 30 époques. Les courbes de précision montrent une montée progressive, avec la précision d'entraînement atteignant environ 0.52, tandis que la précision de validation fluctue autour de 0.45, suggérant que le modèle a des difficultés à généraliser sur les données de validation.

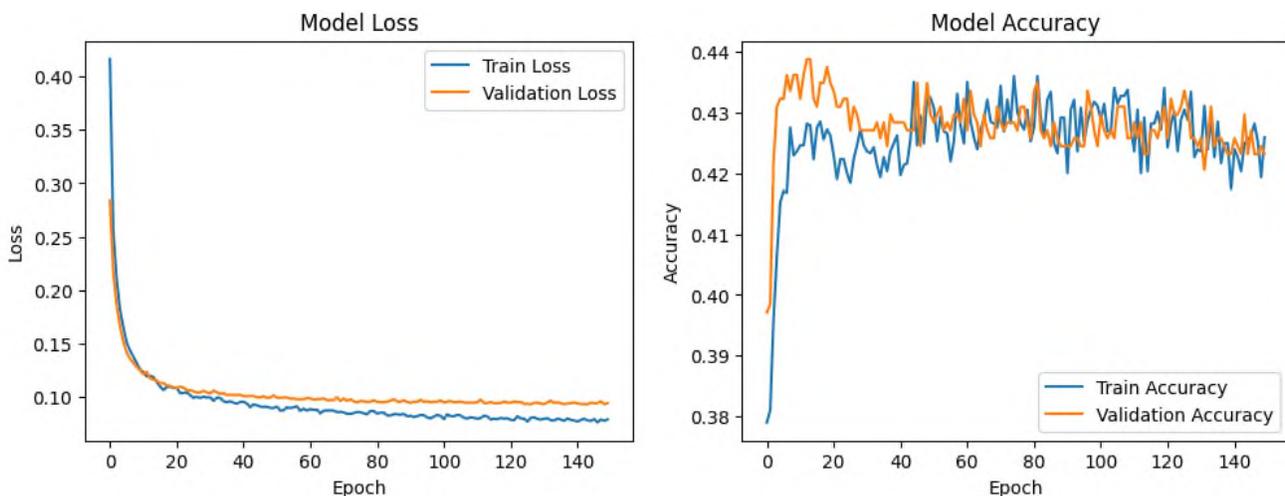


FIGURE 4.3 – Taux d’erreur et précision MLP.

La figure 4.3 présente l'évolution des performances d'un modèle MLP sur 150 époques d'entraînement. Le graphique de gauche montre les variations de la perte pour les ensembles d'entraînement et de validation, tandis que le graphique de droite illustre la précision du modèle sur ces mêmes ensembles.

- **Discussion :** La figure 4.3 illustre l'évolution de la perte et de la précision pour le modèle MLP. Les courbes de perte montrent une diminution rapide au début, suivie d'une stabilisation avec une perte d'entraînement qui diminue légèrement et une perte de validation qui se stabilise autour de 0.10. Les courbes de précision montrent une amélioration progressive, avec la précision d'entraînement et de validation se stabilisant autour de 0.42 à 0.43, mais la précision reste relativement faible, ce qui indique que le modèle peut ne pas être suffisamment complexe pour capturer les patterns des données de manière optimale.

Le tableau 4.1 montre les valeurs des métriques d'évaluation et les taux d'erreur de tous les modèles cités précédemment.

Modèle	Précision	R <sup>2</sup> Score	F1 Score	Rappel	MAE	MSE	RMSE
LSTM	83%	61%	92%	92%	4%	2%	17%
CNN	46%	41%	92%	90%	5%	3%	19%
MLP	40%	48%	61%	60%	5%	3%	17%

TABLE 4.1 – Performance des modèles selon les métriques d'évaluation

- **Discussion :** En termes de précision, le modèle LSTM se distingue avec une précision de 83%, surpassant à la fois le CNN avec 46% et le MLP avec 40%. En ce qui concerne le score  $R^2$ , le LSTM maintient également une avance significative avec un score de 61%, comparé à 41% pour le CNN et 48% pour le MLP. Pour le F1 Score, qui est une mesure de la précision et du rappel, le LSTM obtient un score de 92%, suivi de près par le CNN avec 92% également, et légèrement supérieur au MLP avec 61%. En ce qui concerne le rappel, le LSTM et le CNN maintiennent des performances similaires à 92% et 90% respectivement, tandis que le MLP obtient un score de 60%. En termes d'erreurs, le LSTM montre un MAE de 4%, un MSE de 2%, et un RMSE de 17%, indiquant une prédiction plus précise et cohérente par rapport au CNN et au MLP qui ont des valeurs d'erreur légèrement plus élevées (5% et 3% pour le MAE et le MSE, et 19% pour le RMSE).

En conclusion, le tableau met en évidence que le LSTM surpasse le CNN et le MLP dans la plupart des mesures de performance évaluées. Il démontre une meilleure précision, un meilleur score  $R^2$ , de meilleurs scores F1 et rappel, ainsi que des mesures d'erreur plus faibles. Cela fait du LSTM le choix préféré pour cette tâche spécifique, offrant une performance globale plus fiable comparée aux autres architectures de modèle.

## 4.5.2 Phase de classification

Nous avons utilisé un Random Forest Regressor avec 95 arbres décisionnels ( $n\_estimators=95$ ) et une graine aléatoire fixée à 42 ( $random\_state=42$ ).

### 4.5.2.1 Résultats Obtenus

Dans cette section, nous allons présenter les résultats obtenus après l'entraînement de notre modèle combiné LSTM-RF.

- **Matrice de confusion** : Fournit une vue détaillée des performances du modèle de classification, permettant d'identifier les erreurs de prédiction et d'évaluer la capacité du modèle à différencier les classes. Elle est utilisée pour évaluer et améliorer le modèle de classification. La Matrice de confusion de notre modèle de classification est représentée sur la figure 4.4.

Elle est structurée de la manière suivante : les axes représentent respectivement les classes réelles (axe vertical, "Actual") et les classes prédites par le modèle (axe horizontal, "Predicted"). Les valeurs de la matrice indiquent le nombre d'occurrences pour chaque combinaison de classe réelle et prédite.

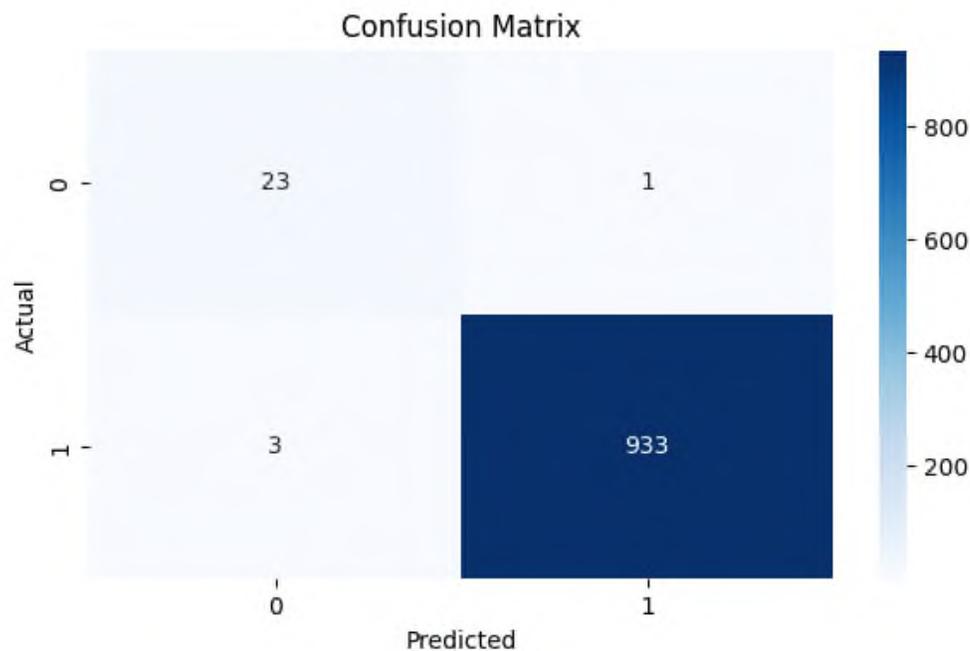


FIGURE 4.4 – La matrice de confusion pour le modèle LSTM-RF.

- **Discussion** : La matrice de confusion présentée montre que notre modèle de classification pour la détection des comportements de conduite dangereux est très performant. Il a correctement identifié 933 cas positifs (1) sur 936, avec seulement 3 faux négatifs. De plus, il a correctement identifié 23 cas négatifs (0) sur 24, avec seulement 1 faux positif. Ces résultats indiquent une haute précision et fiabilité du modèle, avec une précision globale d'environ 99.6%, une précision positive de 99.9%, et un rappel de 99.6%, démontrant ainsi son efficacité pour améliorer la sécurité routière.

Les figures 4.5 et 4.6 représentent des courbes ROC (Receiver Operating Characteristic) pour les classes (normal/anormal).

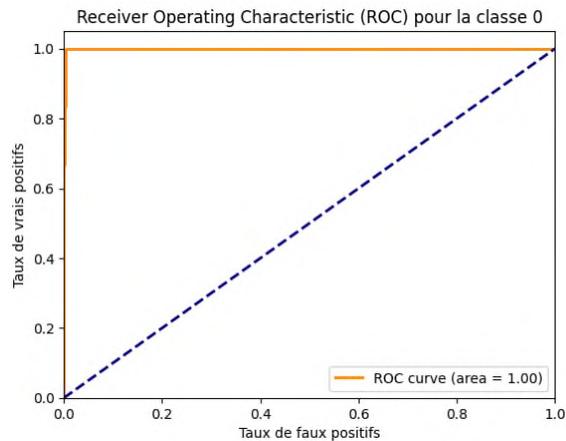


FIGURE 4.5 – La courbe ROC pour la classe 0 (normal).

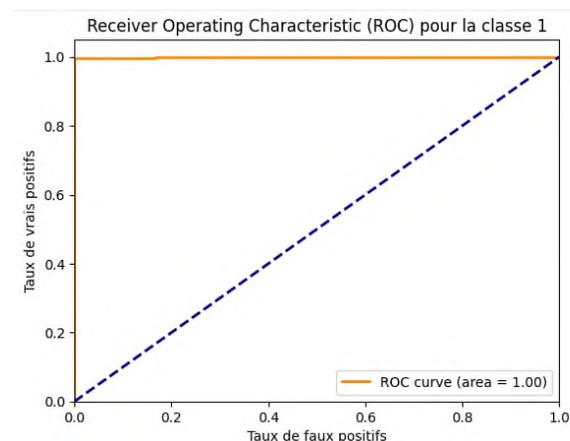


FIGURE 4.6 – La courbe ROC pour la classe 1 (anormal).

- **Discussion :** La courbe orange trace la sensibilité (taux de vrais positifs) par rapport à la spécificité (1 moins le taux de faux positifs), illustrant comment le modèle classe correctement les instances positives tout en minimisant les faux positifs. Pour la classe 0 (normal) dans la figure 4.5 montre que le modèle a une performance parfaite avec une aire sous la courbe (AUC) de 1.00. Cela signifie que pour toutes les valeurs seuils, le modèle peut distinguer parfaitement entre les instances positives et négatives de la classe 0. La courbe passant par le point (0,1) indique que le modèle atteint une sensibilité (taux de vrais positifs) de 100% tout en ayant un taux de faux positifs de 0%, ce qui est idéal. La ligne diagonale en pointillés bleus représente la ligne de hasard ; un modèle performant loin de cette ligne indique une excellente capacité de discrimination. De même, pour la classe 1 (anormal) dans la figure 4.6, la courbe orange présente également une performance parfaite avec une AUC de 1.00. En résumé, ces figures montrent que le modèle a une capacité parfaite à distinguer les classes positives des classes négatives pour les deux classes, ce qui est représenté par des courbes ROC atteignant l'angle supérieur gauche et des AUC de 1.00.

#### 4.5.2.2 Comparaison avec d'autre modèle

Pour évaluer la performance de notre modèle, nous avons effectué une comparaison exhaustive en combinant plusieurs modèles de référence, y compris ceux déjà évalués précédemment pour la phase de détection des comportements dangereux (voir le tableau 4.1). Nous avons notamment comparé le Random Forest à la régression logistique, vu qu'elle est mentionné dans notre état de l'art. Nous présentons un tableau illustratif (voir le tableau 4.2) des résultats après avoir calculé les métriques de performance de chaque modèle, ainsi que les taux d'erreur correspondants.

Modèle	Précision	R <sup>2</sup> Score	F1 Score	Rappel	MAE	MSE	RMSE
LSTM-RF	99%	86%	99%	99%	0.3%	0.3%	6%
CNN-RF	99%	78%	99%	99%	0.5%	0.5%	7%
MLP-RF	98%	48%	98%	98%	33%	16%	4%
LSTM-RL	96%	28%	96%	96%	1%	1%	10%
CNN-RL	98%	52%	98%	98%	1%	1%	10%
MLP-RL	98%	78%	98%	98%	0.5%	0.5%	7%

TABLE 4.2 – Performance des modèles selon les métriques d'évaluation

- Discussion :** Dans le tableau comparatif des cinq modèles, le LSTM-RF se distingue par sa précision exceptionnelle de 99%, surpassant légèrement les autres modèles comme le CNN-RF et le MLP-RF qui affichent également des performances élevées avec des précisions de 99% et 98% respectivement. Le LSTM-RL et le CNN-RL atteignent quant à eux des précisions de 96% et 98%. En termes de score R<sup>2</sup>, le LSTM-RF obtient un score de 86%, ce qui le place en tête par rapport aux autres modèles. Le CNN-RF suit avec un score de 78%, tandis que le MLP-RF, le LSTM-RL et le CNN-RL montrent des scores de 48%, 28% et 52% respectivement. Pour le F1 Score, le LSTM-RF maintient une performance élevée à 99%, suivi de près par le CNN-RF et le MLP-RF qui affichent tous deux 99%. Le LSTM-RL et le CNN-RL obtiennent quant à eux des scores de 96% et 98%. En ce qui concerne le rappel, le LSTM-RF maintient une performance de 99%, suivant de près le CNN-RF et le MLP-RF qui affichent respectivement 99% et 98%. Le LSTM-RL et le CNN-RL montrent des performances de 96% et 98%. En termes de mesures d'erreur, le LSTM-RF se démarque avec un MAE de 0.3%, un MSE de 0.3% et un RMSE de 6%, indiquant une précision notable dans la prédiction des résultats. Comparativement, bien que le CNN-RF et le MLP-RF présentent également de bonnes performances, ils montrent des valeurs légèrement supérieures pour ces métriques d'erreur. En résumé, le LSTM-RF se positionne comme le modèle le plus performant parmi les cinq dans ce tableau, offrant une combinaison impressionnante de précision, de score R<sup>2</sup>, de F1 Score et de mesures d'erreur minimales.

## 4.6 Conclusion

Ce chapitre a détaillé les processus d'expérimentation et d'évaluation de notre modèle de détection des comportements anormaux dans la conduite routière. En utilisant un modèle LSTM pour la détection des anomalies et un classificateur Random Forest pour la classification finale des comportements en tant que normaux ou anormaux, nous avons entrepris une série de tests pour évaluer l'efficacité de notre approche.

Les résultats des expérimentations montrent que le modèle LSTM est particulièrement performant pour capturer les dépendances dans les données de conduite, ce qui est idéal pour identifier les comportements anormaux. La combinaison avec le Random Forest a permis de renforcer la précision de la classification, grâce à sa capacité à gérer les ensembles de données complexes.

# Conclusion générale et perspectives

La détection des comportements de conduite dangereux constitue un enjeu majeur pour améliorer la sécurité routière et optimiser la gestion du trafic. Ce mémoire aborde cette problématique en développant une solution basée sur des techniques avancées d'apprentissage automatique, en particulier les réseaux de neurones récurrents Long Short-Term Memory (LSTM) et les forêts aléatoires (Random Forest). En combinant les forces de ces deux méthodes, nous avons démontré une reconnaissance précise et fiable des comportements de conduite, permettant ainsi une intervention proactive pour réduire les accidents de la route.

Ce travail met en évidence l'efficacité de l'approche hybride LSTM-RF pour la détection des comportements de conduite dangereux et propose une méthodologie complète pour son implémentation. Ces recherches contribuent à l'amélioration de la sécurité routière et offrent des solutions concrètes pour renforcer la sécurité et l'efficacité des systèmes de transport intelligents. En poursuivant sur cette voie, nous espérons voir des avancées continues dans la réduction des accidents de la route et l'amélioration de la mobilité urbaine.

Pour aller plus loin, plusieurs pistes de recherche peuvent être explorées. Tout d'abord, inclure les données des conducteurs, telles que la distraction par l'utilisation de dispositifs électroniques (téléphone, radio) ou par des objets ou des activités dans la voiture (manger, fumer, se maquiller), les mouvements de tête et des yeux (interaction avec les passagers, regard absent ou fixe), ainsi que les signes de stress ou d'agitation (gestes rapides, nerveux) pourrait enrichir le modèle et améliorer la détection des comportements dangereux. De plus, intégrer des données météorologiques telles que les précipitations (pluie, neige), la visibilité (brouillard, brume), la température, et les conditions de la route (verglas, chaussée mouillée) permettrait de prendre en compte les conditions environnementales qui influencent la conduite.

Une autre perspective intéressante serait d'envoyer des signaux d'alerte aux véhicules voisins et aux piétons via des technologies comme les communications V2V (Vehicle-to-Vehicle) et V2P (Vehicle-to-Pedestrian) pour les informer d'un danger imminent. Enfin, inclure dans l'étude les comportements des piétons en interaction avec les conducteurs, tels que la traversée imprévisible de la chaussée, l'utilisation de dispositifs électroniques (téléphone, écouteurs), l'attente aux passages piétons, les gestes indiquant l'intention de traverser, et les réactions aux véhicules approchant, offrirait une vision plus complète et réaliste des scénarios de trafic, contribuant ainsi à une meilleure compréhension et à une gestion plus efficace de la sécurité routière.

# Bibliographie

- [1] AICHA ABBAD. *MASTER EN ÉLECTRONIQUE Spécialité : Systèmes des télécommunications*. PhD thesis, Université Abdelhamid Ibn Badis Mostaganem, 2023.
- [2] Peshawa Jamal Muhammad Ali, Rezhna Hassan Faraj, Erbil Koya, Peshawa J Muhammad Ali, and Rezhna H Faraj. Data normalization and standardization : a technical report. *Mach Learn Tech Rep*, 1(1) :1–6, 2014.
- [3] N Ben Amor, Salem Benferhat, and Zied Elouedi. Réseaux bayésiens naïfs et arbres de décision dans les systèmes de détection d'intrusions. *TSI-Technique et Science Informatiques*, 25(2) :167–196, 2006.
- [4] ABDELFEHTEH BENHAMAHOU. Analyse des paramètres de haralick pour la classification des anomalies mammaires. 2020.
- [5] Azeddine Benlamoudi, Anfal Meriem HADDAOUI, and Yasmina RAHMOUNI. *La Prédiction De La Beauté Faciale Basée Sur Réseaux De Neurone Convolutif*. PhD thesis, UNIVERSITY KASDI MERBAH OUARGLA.
- [6] Mohammed Boughaba, Brahim Boukhris, and MohammedSalim Meflah. L'apprentissage profond (deep learning) pour la classification et la recherche d'images par le contenu. 2017.
- [7] Zakarya Boukemoum. Détection des véhicules par histogramme orienté gradient. 2019.
- [8] Azzedine Boukerche, Yanjie Tao, and Peng Sun. Artificial intelligence-based vehicular traffic flow prediction methods for supporting intelligent transportation systems. *Computer networks*, 182 :107484, 2020.
- [9] Thanina Boultache and Amar Lamzaoui. *Reconnaissance d'activités humaines à l'aide de capteurs de smartphone*. PhD thesis, Université Mouloud Mammeri, 2020.
- [10] Jean Bouyer. La régression logistique en épidémiologie, 2024.
- [11] Jason Brownlee. Roc curves and precision-recall curves for classification in python, 2020. Consulté le 19 juin 2024.
- [12] Zhijun Chen, Chaozhong Wu, Zhen Huang, Nengchao Lyu, Zhaozheng Hu, Ming Zhong, Yang Cheng, and Bin Ran. Dangerous driving behavior detection using video-extracted vehicle trajectory histograms. *Journal of Intelligent Transportation Systems*, 21(5) :409–421, 2017.
- [13] Davit Chicco and Giuseppe Jurman. The coefficient of determination r-squared is more informative than smape, mae, mape, mse and rmse in regression analysis evaluation. *PeerJ Comput. Sci.*, 7 :e623, 2021.
- [14] MOHAMMED ABDELMOUMEN DJABALLAH. Système de prédiction de la consommation d'énergie basé deep learning. 2021.
- [15] En Voiture Simone. Distances de sécurité et temps de réaction, 2024. Consulté le 19 juin 2024.
- [16] Junying Gan, Kaiyong Jiang, Haiying Tan, and Guohui He. Facial beauty prediction based on lighted deep convolution neural network with feature extraction strengthened. *Chinese Journal of Electronics*, 29(2) :312–321, 2020.

- [17] DJALLEL HAMOUDA. Un système de détection d'intrusion pour la cybersécurité. 2020.
- [18] Jie Hu, Li Xu, Xin He, and Wuqiang Meng. Abnormal driving detection based on normalized driving behavior. *IEEE Transactions on Vehicular Technology*, 66(8) :6645–6652, 2017.
- [19] SAADI IBTISSAM. La prédiction de flux de trafic routier par une méthode d'apprentissage profond. 2020.
- [20] Muhamad Irsan, Rosilah Hassan, Mohammad Khatim Hasan, and Lam Meng Chun. Future of transport : A framework for smart driver. In *2019 Symposium on Future Telecommunication Technologies (SOFTT)*, volume 1, pages 1–5. IEEE, 2019.
- [21] Landu Jiang, Wen Xie, Dian Zhang, and Tao Gu. Smart diagnosis : Deep learning boosted driver inattention detection and abnormal driving prediction. *IEEE Internet of Things Journal*, 9(6) :4076–4089, 2021.
- [22] Chris Jing. Application of random forest-based supervised ensemble learning method for hail nowcasting in the midwestern united states. 2024.
- [23] Sabri Khamari. *Architectures et protocoles pour les véhicules connectés*. PhD thesis, Bordeaux, 2023.
- [24] Emanuele Lattanzi and Valerio Freschi. Machine learning techniques to identify unsafe driving behavior by means of in-vehicle sensor data. *Expert Systems with Applications*, 176 :114818, 2021.
- [25] LuoJun Lin, Lingyu Liang, Lianwen Jin, and Weijie Chen. Attribute-aware convolutional neural networks for facial beauty prediction. In *IJCAI*, pages 847–853, 2019.
- [26] Jia Liu, Wei Huang, Hao Li, Shengong Ji, Yajun Du, and Tianrui Li. Slafusion : Attention fusion based on sax and lstm for dangerous driving behavior detection. *Information Sciences*, 640 :119063, 2023.
- [27] Jia Liu, Tianrui Li, Zhong Yuan, Wei Huang, Peng Xie, and Qianqian Huang. Symbolic aggregate approximation based data fusion model for dangerous driving behavior detection. *Information Sciences*, 609 :626–643, 2022.
- [28] Athanasios Maimaris and George Papageorgiou. A review of intelligent transportation systems from a communications technology perspective. In *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, pages 54–59. IEEE, 2016.
- [29] Messaoud Mezati and Radja Soundes Otmani. *Réalisation d'une application intelligente pour l'ambulance (Smart Ambulance)*. PhD thesis, UNIVERSITY OF KASDI MERBAH OUARGLA.
- [30] Michael Motro, Alice Chu, Junil Choi, Patricia S Lavieri, Abdul Rawoof Pinjari, Chandra R Bhat, Joydeep Ghosh, and Robert W Heath Jr. Vehicular ad-hoc network simulations of overtaking maneuvers on two-lane rural highways. *Transportation research part C : emerging technologies*, 72 :60–76, 2016.
- [31] Ahmed Malek Nada. L'intelligence ambiante et les systèmes de transport intelligents. *Mémoire du Diplôme Magister, Université Badji Mokhtar Annaba*, 2014.
- [32] Commission nationale pour la protection des données (CNPD). Intelligence artificielle, 2023. Consulté le : 12 juin 2024.
- [33] OpenClassrooms. Évaluez les performances d'un modèle de machine learning : Évaluez un algorithme de régression, 2024. Accessed : 2024-06-17.
- [34] Mehdi Othmani-Guibourg, Amal El Fallah-Seghrouchni, and Jean-Loup Farges. Lstm path-maker : une nouvelle stratégie pour la patrouille multiagent basée sur l'architecture lstm. In *Conférence Nationale en Intelligence Artificielle*, 2019.

- [35] Eduardo Romera, Luis M Bergasa, and Roberto Arroyo. Need data for driver behaviour analysis? presenting the public uah-driveset. In *2016 IEEE 19th international conference on intelligent transportation systems (ITSC)*, pages 387–392. IEEE, 2016.
- [36] Roole. Distance de sécurité sur la route : comment la calculer?, 2024. Consulté le 19 juin 2024.
- [37] Cian Ryan, Finbarr Murphy, and Martin Mullins. End-to-end autonomous driving risk analysis : A behavioural anomaly detection approach. *IEEE Transactions on Intelligent Transportation Systems*, 22(3) :1650–1662, 2020.
- [38] Mohammad Shahverdy, Mahmood Fathy, Reza Berangi, and Mohammad Sabokrou. Driver behavior detection and classification using deep convolutional neural networks. *Expert Systems with Applications*, 149 :113240, 2020.
- [39] Wei Song, Guangde Zhang, and Yicheng Long. Identification of dangerous driving state based on lightweight deep learning model. *Computers and Electrical Engineering*, 105 :108509, 2023.
- [40] Romain Tavenard. Réseaux neuronaux récurrents, 2023. Consulté le : 12 juin 2024.
- [41] Arturo Tejada, Jeroen Manders, Ron Snijders, Jan-Pieter Paardekooper, and Stefanie de Hair-Buijssen. Towards a characterization of safe driving behavior for automated vehicles based on models of “typical” human driving behavior. In *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, pages 1–6. IEEE, 2020.
- [42] Yan Tian, Kaili Zhang, Jianyuan Li, Xianxuan Lin, and Bailin Yang. Lstm-based traffic flow prediction with missing data. *Neurocomputing*, 318 :297–305, 2018.
- [43] Patara Trirat and Jae-Gil Lee. Df-tar : a deep fusion network for citywide traffic accident risk prediction with dangerous driving behavior. In *Proceedings of the Web Conference 2021*, pages 1146–1156, 2021.
- [44] Unknown. Exemples de systèmes de transport intelligents, 2023. Consulté le : 12 juin 2024.
- [45] Xinrong Wu, Junwei Zhou, Jinghe An, and Yanchao Yang. Abnormal driving behavior detection for bus based on the bayesian classifier. In *2018 Tenth International Conference on Advanced Computational Intelligence (ICACI)*, pages 266–272. IEEE, 2018.
- [46] Lu Xu, Jinhai Xiang, and Xiaohui Yuan. Crnet : Classification and regression neural network for facial beauty prediction. In *Pacific Rim Conference on Multimedia*, pages 661–671. Springer, 2018.
- [47] Tianbao Ying and Yiming Ying. Auc maximization in the era of big data and ai : A survey. *ACM Computing Surveys*, 55(8) :1–37, 2022.
- [48] Jiadi Yu, Zhongyang Chen, Yanmin Zhu, Yingying Chen, Linghe Kong, and Minglu Li. Fine-grained abnormal driving behaviors detection and identification with smartphones. *IEEE transactions on mobile computing*, 16(8) :2198–2212, 2016.
- [49] Wenhao Yu and Qinghong Huang. A deep encoder-decoder network for anomaly detection in driving trajectory behavior under spatio-temporal context. *International Journal of Applied Earth Observation and Geoinformation*, 115 :103115, 2022.
- [50] Gizem Yükses et al. Mémoire de fin d’études : " l’intelligence artificielle" text-to-image" comme outils de support à l’idéation en conception architecturale.". 2023.
- [51] Lan Zhang, Li Yan, Yuguang Fang, Xuming Fang, and Xiaoxia Huang. A machine learning-based defensive alerting system against reckless driving in vehicular networks. *IEEE Transactions on Vehicular Technology*, 68(12) :12227–12238, 2019.

- 
- [52] Weiguo Zhang and Yunxia Xiao. Detection of dangerous driving behavior based on cg-yolov5. In *2022 IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC)*, pages 1–6. IEEE, 2022.
- [53] Li Zhu, Fei Richard Yu, Yige Wang, Bin Ning, and Tao Tang. Big data analytics in intelligent transportation systems : A sur-vey. *IEEE Transactions on Intelligent Transportation Systems*, 20(1) :383–398, 2018.

## Résumé

Les systèmes de transport intelligents (STI) ont révolutionné la mobilité urbaine grâce à l'intégration de technologies avancées telles que les capteurs, les communications sans fil et l'intelligence artificielle. Cependant, la sécurité routière reste une préoccupation majeure, les comportements de conduite inappropriés constituant une menace constante. Pour relever ce défi, notre étude propose une solution basée sur des techniques avancées d'apprentissage automatique, utilisant les réseaux de neurones récurrents LSTM et les forêts aléatoires (RF) pour détecter de manière précise et fiable les comportements de conduite dangereux. Nos résultats montrent que l'approche hybride LSTM-RF surpasse les autres modèles en minimisant les faux positifs et les faux négatifs, augmentant ainsi la fiabilité de la détection des comportements dangereux. En intégrant les capacités de détection des LSTM avec l'efficacité de classification des RF, notre méthode contribue à la sécurité routière et à l'amélioration des STI.

**Mots clés :** STI, sécurité routière, LSTM, forêts aléatoires, approche hybride LSTM-RF.

## Abstract

Intelligent Transportation Systems (ITS) have revolutionized urban mobility through the integration of advanced technologies such as sensors, wireless communications, and artificial intelligence. However, road safety remains a major concern, with inappropriate driving behaviors posing a constant threat. To address this challenge, our study proposes a solution based on advanced machine learning techniques, utilizing Long Short-Term Memory (LSTM) recurrent neural networks and Random Forests (RF) to accurately and reliably detect dangerous driving behaviors. Our results show that the hybrid LSTM-RF approach outperforms other models by minimizing false positives and false negatives, thereby increasing the reliability of dangerous behavior detection. By integrating the detection capabilities of LSTMs with the classification efficiency of RFs, our method contributes to road safety and the improvement of ITS.

**Keywords :** ITS, Road Safety, LSTM, Random Forests, hybrid LSTM-RF approach.