

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
Université A. Mira de Béjaïa  
Faculté des Sciences Exactes  
Département d'Informatique



## Mémoire de Fin d'étude

En vue de l'obtention du diplôme de Master Professionnel en Informatique

Option : Administration et Sécurité des Réseaux

Thème

---

# Amélioration de la sûreté dans les systèmes de transports intelligents

---

Réalisé par

M<sup>lle</sup> ACHOUR Romissa

M<sup>lle</sup> AISSANI Lina

Devant le jury composé de

<b>Président :</b>	Mr. TOUAZI Djoudi	M.C.B	Université de Béjaïa
<b>Examineur :</b>	M <sup>me</sup> . CHABANNE Sarah	M.C.B	Université de Béjaïa
<b>Encadrant :</b>	Mr. AISSANI Sofiane	M.C.A	Université de Béjaïa
<b>Co-Encadrante :</b>	M <sup>me</sup> ZIZI Kahina	Doctorante	Université de Béjaïa
<b>Invitée :</b>	M <sup>lle</sup> CHENACHE Tinhinane	Doctorante	Université 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 pour nous avoir donné la force, la volonté et le courage nécessaires pour mener à bien ce travail.

Nous exprimons notre gratitude envers notre encadrant, **M. AISSANI Sofiane**, pour son soutien constant, ses conseils avisés et sa confiance pour nous avoir accompagné.

Nous tenons à remercier très chaleureusement notre Co-ecadrante Mme **ZIZI Kahina** pour sa collaboration précieuse et ses encouragements tout au long de la rédaction de ce mémoire, ainsi qu'un grand merci à Melle **CHENACHE Tinhinane** pour sa participation.

Nous adressons nos remerciements aux membres du jury, M. **TOUAZI Djoudi** et Mme **CHABANNE Sarah**, pour avoir consacré leur temps à l'évaluation de ce travail.

Nos remerciements les plus sincères vont **à nos parents**, dont le soutien indéfectible a été une source d'inspiration et de motivation tout au long de cette aventure.

Enfin, nous tenons à remercier chaleureusement tous les membres du département d'Informatique de l'Université de Béjaïa pour leur soutien et leur contribution à notre formation.

---

# Dédicaces

---

*Je dédie ce mémoire*

**A mes chers parents** que nulle dédicace ne puisse exprimer mes sincères sentiments et mon éternelle gratitude, pour leur patience illimitée, leurs encouragements continus, leur aide, en témoignage de mon profond amour et respect pour leurs grands sacrifices

**A ma soeur : Lina (Dida)**

**A mes frères : Badis et Amazigh**

Sans oublier **ma grand mère : Baya, et mon grand père :salah** et mes tantes : **Khoukha,Zohra,Hakima,Oahiba,Sabrina** et sans oublier mes oncles : **Madjed , Ralime** A mes amies et amis exceptionnellement : **Fatima,Feriel,Massilia, Melissa, Lina, Selma,Ines, Maya, Lissa, Leticia, Youcef, Ikram.**

**A mon encadreur M. AISSANI Sofiane**

et à **ma Co-encadrante Mme ZIZI Kahina** et tous ceux qui me connaissent de loin ou de près.

*Romaissa*

---

# Dédicaces

---

*Je dédie ce mémoire*

**A mes chers parents** que nulle dédicace ne puisse exprimer mes sincères sentiments et mon éternelle gratitude, pour leur patience illimitée, leurs encouragements continus, leur aide, en témoignage de mon profond amour et respect pour leurs grands sacrifices

**A mes sœurs : Litissia, Céline, Lilia et Rania**

Sans oublier **Mes chères cousines : Kahina et Dyhia.** et **mes chers grands parents et mes oncles**

A mes amies exceptionnellement : **Milane, Salema, Melissa, Romaisa .**

**A mon encadreur M. AISSANI Sofiane**

et à **ma Co-encadrante Mme ZIZI Kahina** et tous ceux qui me connaissent de loin ou de près.

*Lina*

# Table des matières

Liste des figures	iv
Liste des tableaux	v
Liste des abréviations	vi
Introduction générale	1
<b>1 Pré-requis théoriques</b>	<b>3</b>
1.1 Introduction	3
1.2 Système de transport intelligent (STIs)	3
1.2.1 Définition des STIs	3
1.2.2 L'évaluation	4
1.2.3 Les avantages et les inconvénients des STIs	4
1.2.4 Categories et domaines d'application des STIs	5
1.3 Gestion et controle du trafic	6
1.3.1 Problèmes liés au trafic dans les STIs	6
1.3.2 Nature et causes de la congestion routière dans les zones urbaines	7
1.4 Apprentissage automatique et systeme transport intelligent :	8
1.4.1 Les types d'apprentissage automatique	8
1.4.2 L'apprentissage profond (DEEP LEARNING)	9
1.4.3 Réseaux de neurones récurrents (RNN)	10
1.4.4 Intégration de l'Apprentissage Automatique dans les STIs	11
1.5 Conclusion	13
<b>2 Etat de l'art sur la prédiction et la détection de l'état du trafic</b>	<b>14</b>
2.1 Introduction	14
2.2 Classification des travaux passés en revus	14
2.3 Etude des travaux existants de trafic routir	15
2.3.1 Solutions basées sur la détection de l'état de trafic	15
2.3.2 Solutions basées sur la prédiction de l'état de trafic	16
2.3.3 Solutions basées sur la prédiction et la détection de l'état de trafic	19
2.4 Etude comparative	23
2.4.1 Critères de comparaison	23
2.4.2 Tableau comparatif	24

2.4.3	Discussion . . . . .	25
2.5	Conclusion . . . . .	26
<b>3</b>	<b>Système de Prédiction et Détection de congestion (Bi-ConvLSTM)</b>	<b>27</b>
3.1	Introduction . . . . .	27
3.2	Motivation . . . . .	27
3.3	Conception générale du BiConvLSTM . . . . .	28
3.4	Conception détaillée du Bi-ConvLSTM . . . . .	29
3.4.1	Identification de l'ensemble de données . . . . .	30
3.4.2	Découpage et analyse de dataset réelles . . . . .	32
3.4.3	Pré-traitement et sélection des caractéristiques . . . . .	33
3.4.4	Modélisation et entraînement du modèle . . . . .	37
3.4.5	Test et évaluation du modèle . . . . .	41
3.5	Conclusion . . . . .	43
<b>4</b>	<b>Simulation et Evaluation de Performances</b>	<b>44</b>
4.1	Introduction . . . . .	44
4.2	Environnement de travail . . . . .	44
4.3	Outils et bibliothèques utilisés . . . . .	44
4.4	Simulation et résultats : . . . . .	46
4.4.1	Hyperparamètres : . . . . .	47
4.4.2	Résultats obtenus : . . . . .	47
4.5	Étude Comparative . . . . .	50
4.5.1	Comparaison selon le taux d'erreur (Loss) . . . . .	50
4.5.2	Comparaison selon les performances des modèles . . . . .	52
4.6	Conclusion . . . . .	54
	<b>Conclusion générale et perspectives</b>	<b>55</b>
	<b>Bibliographie</b>	<b>57</b>

# Table des figures

1.1	l'apprentissage automatique[1]. . . . .	8
1.2	Structure d'un réseaux de neurones récurrents (RNN)[2]. . . . .	10
2.1	Classification des travaux passés en revus. . . . .	15
3.1	L'architecture générale du Bi-ConvLSTM. . . . .	29
3.2	Architecture détaillée du Bi-ConvLSTM. . . . .	30
3.3	Le segment d'étude. . . . .	31
3.4	Modèle Bi-ConvLstm . . . . .	38
3.5	Architecture du modèle Bi-ConvLSTM . . . . .	40
4.1	la variation de l'erreur (Loss) du notre modèle Bi-ConvLSTM. . . . .	48
4.2	La variation de l'erreur (Loss) des deux modèles. . . . .	51
4.3	Diagramme combiné des métriques des différents modèles . . . . .	52

# Liste des tableaux

2.1	Tableau comparatif. . . . .	24
3.1	la description des champs de dataset. . . . .	32
4.1	Hyperparamètres du modèle . . . . .	47
4.2	Résultats des métriques d'évaluation . . . . .	49
4.3	Comparaison des Métriques pour différents modèles . . . . .	52



# Liste des abréviations

<b>ANN</b>	Artificial Neural Network.
<b>BLSTME</b>	BiradctionnelLong Short-Term Memory .
<b>BPTT</b>	Rétro-propagation à travers le temps.
<b>CMTF</b>	Critical Manufacturing Task Force.
<b>CNN</b>	Convolutional Neural Network.
<b>DL</b>	Deep learning.
<b>GPS</b>	Global Positioning System.
<b>GRU</b>	Gated Recurrent Unit
<b>IA</b>	Artificial Intelligence .
<b>LSTM</b>	Long Short-Term Memory.
<b>MAE</b>	Mean Absolute Error.
<b>ML</b>	Machine Learnin.
<b>MRE</b>	l'erreur relative moyenne.
<b>NTP</b>	Network Time Protocol.
<b>PCNN</b>	Deep Convolutional Networks for Short-Term Traffic Congestion Prediction.
<b>RCP</b>	Remote Control Protocol.
<b>RF</b>	Random Fores.
<b>RMSE</b>	Root Mean Square Error
<b>RNA</b>	Réseau de neurones artificie.
<b>RNN</b>	Réseaux de neurones récurrent
<b>RSU</b>	les unités en bord de route .
<b>SIG</b>	Système d'Information Géographique.
<b>STI</b>	Système de Transport Intelligent.
<b>V2R</b>	Vehicle-to-Road
<b>V2V</b>	Vehicle-to-Vehicle

# Introduction générale

De nos jours, la congestion routière et les embouteillages sont devenus un problème mondial majeur, engendrant des pertes économiques considérables et des répercussions néfastes sur l'environnement. Avec l'augmentation rapide du nombre de véhicules sur les routes, les villes font face à des niveaux de trafic sans précédent. Les temps de trajet sont prolongés en raison de cette situation, ce qui diminue la productivité et accroît le stress des conducteurs. Devant cette augmentation continue de la congestion routière, il est essentiel d'utiliser des solutions technologiques pour améliorer la gestion du trafic urbain.

La prédiction de l'état du trafic est l'une des nombreuses applications des Systèmes de Transport Intelligents (ITS). Diverses technologies de détection et de surveillance du trafic sont intégrées dans les ITS afin de recueillir des informations en temps réel sur l'état des routes. Le trafic est surveillé par des capteurs, des caméras de surveillance et des systèmes de positionnement global (GPS) afin de détecter les incidents et d'évaluer la densité du trafic.

Par la suite, des algorithmes d'intelligence artificielle (IA) sont utilisés pour traiter ces informations afin de produire des prédictions précises sur les conditions de trafic à venir, en général, ces algorithmes sont appelés d'apprentissage profond et reposent sur les réseaux de neurones (NN), notamment les réseaux de neurones récurrents (RNN).

Le travail de ce mémoire s'inscrit dans le contexte de la conception et du développement d'un système de détection et prédiction de l'état de trafic visant à orienter les conducteurs pour une circulation optimale, ce qui améliore la sécurité routière et le confort des usagers. Grâce à des prévisions précises, le système permet d'éviter les embouteillages et de sélectionner les itinéraires les plus performants.

Ainsi, nous choisirons une approche d'apprentissage profond provenant de l'intelligence artificielle, qui repose sur un réseau de neurones récurrents, en particulier le RNN de type mémoire à long terme (LSTM). La méthode comprend deux modules : une mémoire convolutive à long terme (ConvLSTM2D) qui sert d'encodeur pour extraire les caractéristiques spatio-temporelles, et une mémoire bidirectionnelle LSTM (Bi-LSTM) qui sert de décodeur pour extraire les caractéristiques

périodiques des flux de trafic.

Ce mémoire est organisé en quatre chapitres. Le premier chapitre portera sur des généralités des systèmes de Transport Intelligents, l'Internet des véhicules et ses applications, les problèmes liés au trafic routier, et l'apprentissage machine, ses différents types et leur phase d'apprentissage ainsi que les modèles à base d'apprentissage pour la prédiction de flux de trafic. Le deuxième chapitre présente un état de l'art sur l'apprentissage profond pour la détection et la prédiction de l'état du trafic, ainsi qu'une comparaison de ses différentes approches. Dans le troisième chapitre, nous exposerons la conception de notre système de prédiction, à savoir sa conception globale et détaillée, ainsi que les améliorations envisagées. Dans le quatrième chapitre, nous présenterons la mise en place de notre système, en examinant les environnements matériels et logiciels utilisés pour le développement de notre système, ainsi que les diverses expériences et les résultats obtenus. Une conclusion suivie de perspectives clôtureront notre mémoire.

# Chapitre 1

## Pré-requis théoriques

### 1.1 Introduction

Les systèmes de transport intelligents (STI) sont des applications avancées qui intègrent des technologies émergentes pour améliorer la gestion, l'efficacité et la sécurité des systèmes de transport. Ces systèmes tirent parti de capacités associées à l'intelligence, telles que la communication, le traitement de l'information et le comportement adaptatif, pour optimiser l'utilisation des infrastructures de transport, renforcer la sécurité routière, et favoriser le développement durable. Les STI jouent un rôle crucial dans la gestion du trafic, la sécurité des déplacements, et la réduction des congestions, en s'appuyant sur des technologies telles que les capteurs, les communications avancées, et les systèmes de gestion de l'information. En constante évolution, les STI offrent des avantages significatifs en termes d'efficacité, de sécurité et de coûts pour les transports publics et les déplacements urbains, contribuant ainsi à l'émergence de villes intelligentes et à l'amélioration de la qualité de vie des citoyens.

### 1.2 Système de transport intelligent (STIs)

#### 1.2.1 Définition des STIs

Depuis le milieu des années 60, le système de transport intelligent (STI) est un domaine en plein essor qui a connu une évolution continue au fil du temps. Il s'agit d'un système global de gestion des services de transport, qualifié d'intelligent car il repose sur des fonctions liées à l'intelligence, telles que le traitement de l'information, permettant ainsi de traiter, d'analyser et de partager des informations liées à ce service[3]. Aujourd'hui, il utilise les technologies avancées que le développement et la recherche nous ont apportées, telles que les capteurs, les réseaux de communication et l'analyse des données, pour améliorer l'efficacité, la sécurité et la durabilité des systèmes de transport. Les ITS visent à améliorer et à fluidifier le système de transport, ce qui

permet de réduire les congestions et d'optimiser l'utilisation des ressources routières.

### 1.2.2 L'évaluation

L'évaluation des Systèmes de Transport Intelligents (STI) pour le trafic routier repose sur plusieurs critères clés : efficacité opérationnelle, sécurité, et impact environnemental. L'efficacité opérationnelle est mesurée par la capacité des STI à réduire les temps de trajet et à améliorer la fluidité du trafic en utilisant des technologies comme les feux de signalisation adaptatifs et les systèmes de gestion du trafic en temps réel[4]. En termes de sécurité, les STI peuvent diminuer les accidents grâce à des avertissements de collision et des systèmes d'alerte précoce[5]. Enfin, l'impact environnemental est évalué en analysant la réduction des émissions de gaz à effet de serre et la consommation d'énergie grâce à une meilleure gestion du trafic et à des véhicules plus efficaces[6]. Les études montrent que l'intégration de ces systèmes permet d'optimiser les ressources existantes et de promouvoir une mobilité plus durable et sécurisée[7].

### 1.2.3 Les avantages et les inconvénients des STIs

Grâce à son utilisation des technologies de l'information, les Systèmes de Transport Intelligents (STI) offrent une multitude d'avantages significatifs pour améliorer l'efficacité globale du réseau de transport [8].

Les Systèmes de Transport Intelligents (STI) présentent une gamme d'avantages significatifs, grâce à leur utilisation des technologies de l'information. Tout d'abord, ils contribuent à améliorer la mobilité en facilitant les changements dans les déplacements et le mode de vie des individus. De plus, les STI aident à réduire la congestion du trafic en fournissant des informations en temps réel sur les conditions routières, permettant ainsi aux utilisateurs de choisir des itinéraires plus efficaces [8][9]. En outre, ces systèmes contribuent à la prévention des accidents en fournissant des avertissements sur les dangers potentiels et les conditions météorologiques [10].

Cependant, malgré ces avantages, les STI présentent également des inconvénients. Tout d'abord, leur mise en œuvre nécessite des investissements importants en termes d'infrastructures, d'équipements et de technologies, rendant leur accessibilité limitée à certains segments de la population [8]. De plus, leur efficacité est limitée dans les zones sans connexion Internet, et de nombreux véhicules ne disposent pas des technologies nécessaires pour les utiliser pleinement [9]. De plus, les STI nécessitent des ressources pour fonctionner correctement, et leur manque de ressources peut entraver leur efficacité. Enfin, en raison du manque de sensibilisation, de nombreuses personnes ne sont pas conscientes des avantages des STI, ce qui souligne la nécessité d'une sensibilisation accrue pour promouvoir leur adoption et leur utilisation efficace [10].

## 1.2.4 Catégories et domaines d'application des STIs

### 1.2.4.1 Catégorisation des applications relatives aux STIs

L'intégration des Systèmes de Transport Intelligents (STI) se décline dans divers domaines, des préparatifs de voyage jusqu'à la gestion des urgences et au paiement électronique. Ces systèmes offrent une multitude de services, tels que des informations pour les voyageurs sur la préparation au voyage, les alertes en cours de trajet pour les conducteurs et les passagers des transports publics, ainsi que des données sur les conditions routières et la localisation. En matière de gestion du trafic, les STI interviennent dans le contrôle du trafic, la gestion des accidents et des flux, ainsi que dans la planification et la maintenance des infrastructures. Pour les véhicules, ils permettent des améliorations telles que l'automatisation des fonctions de conduite, les systèmes d'évitement des collisions et la maintenance préventive. De même, dans le domaine des véhicules commerciaux, les STI facilitent les formalités administratives, la sécurité routière et la gestion des parcs automobiles[11]. La gestion des transports publics, des urgences et la sécurité des voyageurs sont également des aspects clés pris en charge par ces systèmes. En outre, les STI contribuent à une utilisation plus durable des itinéraires grâce à une meilleure coordination des déplacements, basée sur des données en temps réel, ce qui favorise une planification efficace[12].

### 1.2.4.2 Domaines d'application des STIs

Les Systèmes de Transport Intelligents (STIs) peuvent apporter des solutions à divers défis spécifiques du domaine de la mobilité. Plus largement, ils ont le potentiel d'améliorer les performances des transports en respectant les critères du développement durable, notamment sur les plans environnemental, sécuritaire, de la gestion de la congestion, de la productivité et de l'efficacité. De plus, ils peuvent améliorer l'information des voyageurs et leur confort[12].

Les Systèmes de Transport Intelligents (STIs) améliorent la sécurité en rendant les transports plus sûrs et en réduisant les impacts des catastrophes naturelles ou humaines. Grâce à une planification efficace, ils permettent de réduire les délais d'intervention des services d'urgence et de créer des itinéraires d'évacuation prioritaires[12]. Les STIs contribuent également à une réduction durable des accidents en signalant les conditions dangereuses aux voyageurs. En matière de gestion de la congestion, les STIs augmentent l'efficacité des réseaux de transport en temps réel grâce à des capteurs et autres instruments. Ils optimisent l'exploitation des systèmes de transport, réduisant ainsi la congestion. Sur le plan de la productivité et de l'efficacité, les STIs améliorent les opérations de transport en gérant les parcs de véhicules, réduisant les coûts administratifs et opérationnels, et augmentant considérablement la productivité grâce à des calculs fiables des temps de déplacement et des livraisons en juste à temps[13][12].

## 1.3 Gestion et controle du trafic

Pour une efficacité maximale dans l'utilisation des routes et des services routiers, il ne suffit pas de disposer de véhicules intelligents si les routes elles-mêmes sont mal entretenues ou congestionnées. Les embouteillages peuvent entraîner des blocages partiels ou totaux, affectant tous les modes de transport. Les Systèmes de Transport Intelligents (STIs) jouent un rôle essentiel dans la gestion du trafic routier grâce aux moyens et équipements dynamiques utilisés par les postes de contrôle de circulation sur les réseaux routiers et autoroutiers. Ils représentent également un atout majeur pour l'information des usagers et, plus généralement, pour l'exploitation des routes[14].

La connaissance du trafic est un élément essentiel pour l'exploitation du réseau routier, tant en temps réel, pour alimenter les systèmes d'aide à la gestion du trafic, qu'en temps différé, pour éclairer les politiques publiques et fournir des statistiques sectorielles[14]. Les données de trafic sont utilisées pour élaborer la politique de gestion, d'exploitation et d'entretien du réseau routier national, suivre l'évolution du trafic routier et son impact environnemental, optimiser l'utilisation des infrastructures existantes conformément aux objectifs du Grenelle de l'Environnement et du Schéma National des Infrastructures de Transport, élaborer et évaluer la politique publique de réduction de l'insécurité routière, et prévoir le trafic à court, moyen et long terme, notamment avec le calendrier Bison Futé et les modèles de prévision de trafic pour l'évaluation ex-ante des projets d'infrastructures. Les utilisateurs de ces données sont nombreux et variés, incluant les directions d'administration centrale, les directions interdépartementales des routes (DIR), les directions régionales de l'environnement, de l'aménagement et du logement, ainsi que les organismes du réseau scientifique et technique au sein du ministère. Les gestionnaires routiers hors État, tels que les conseils départementaux, les métropoles, et les bureaux d'études, utilisent également ces données.

### 1.3.1 Problèmes liés au trafic dans les STIs

La gestion des flux de véhicules dans une ville est une problématique importante qui donne lieu à de nombreuses études.une augmentation du trafic de véhicules ressemble à un énorme casse-tête qui rendre nos trajets stressant .Elle implique la mise en place de stratégies et de technologies visant à réguler le flux de véhicules, à prévenir les embouteillages, à optimiser l'utilisation des infrastructures routières et à garantir la sécurité des usagers.

#### 1.3.1.1 Sécurité routière :

Malgré les systèmes de transport intelligent, les accidents de la route peuvent toujours survenir en raison de comportements imprudents des conducteurs, de conditions météorologiques défavorables ou de défaillances mécaniques[3].

Les STI sont également axés sur l'amélioration de la sécurité routière en fournissant des informations en temps réel sur les conditions routières dangereuses, en encourageant les comportements

de conduite sûrs grâce à des systèmes d’alerte et de prévention des collisions, et en facilitant la coordination des services d’urgence en cas d’accident[8].

### 1.3.1.2 Gestion inefficace des flux de circulation :

Les STI sont également axés sur l’amélioration de la sécurité routière en fournissant des informations en temps réel sur les conditions routières dangereuses, en encourageant les comportements de conduite sûrs grâce à des systèmes d’alerte et de prévention des collisions, et en facilitant la coordination des services d’urgence en cas d’accident[3].

### 1.3.1.3 Congestion Routière :

La congestion du trafic est l’un des problèmes les plus préoccupants du monde moderne, qui désire satisfaire le besoin de mobilité sans entrave. Dans la zone métropolitaine, la situation est définie comme congestion lorsque la vitesse moyenne est inférieure de plus de 50% à la vitesse en situation d’écoulement libre, et le terme de congestion désigne la dégradation de la qualité de service quand le nombre d’usagers augmente[15]. Ce phénomène se caractérise par l’apparition de retards, voire de goulets d’étranglement en période de fort trafic, c’est-à-dire quand la capacité de l’infrastructure devient insuffisante pour réguler les flux. Le problème est fréquent localement et périodiquement, notamment dans les grandes villes et lors des grands départs pour les vacances.

## 1.3.2 Nature et causes de la congestion routière dans les zones urbaines

### — Congestion récurrente :

La congestion récurrente se produit principalement lorsqu’un trop grand nombre de véhicules utilisent le réseau routier en même temps, Généralement aux heures de pointe du matin et de l’après-midi en semaine lorsque la vaste majorité des gens se rendent au travail ou à l’école et reviennent à la maison aux mêmes heures. Dans les zones urbaines plus petites, la période de pointe s’étend normalement sur une période plus courte (une heure ou deux)[16].

### — Congestion incidente :

La congestion incidente associe à des conditions aléatoires ou spéciales et à des événements particuliers[16], les quatre causes principales de la congestion incidente sont :

- Les incidents de la circulation (les pannes, les accidents graves).
- Les chantiers de construction.
- Les conditions météorologiques.
- Les événements spéciaux.

En raison du caractère aléatoire de ce type de congestion, elle est plus difficile à prévoir, à quantifier et à gérer . Le manque de données sur les incidents et leur impact sur les temps



de déplacement a mené certains organismes à recourir à des estimations provisoires de la congestion incidente.

## 1.4 Apprentissage automatique et systeme transport intelligent :

L'apprentissage automatique (Machine Learning -ML-) est une branche de l'intelligence artificielle .C'est une sphère d'étude où les ordinateurs sont dotés de la capacité d'assimiler des connaissances et de s'améliorer sans qu'il soit nécessaire de leur enseigner chaque détail de manière explicite [17].

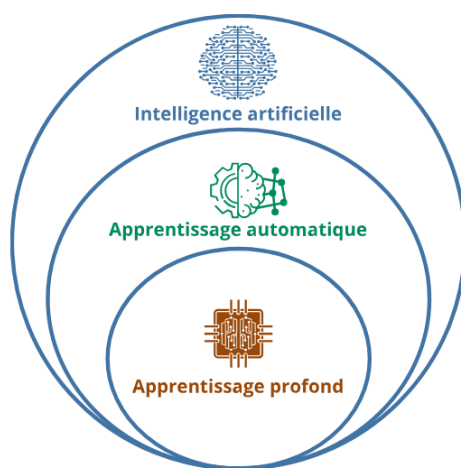


FIGURE 1.1 – l'apprentissage automatique[1].

### 1.4.1 Les types d'apprentissage automatique

#### 1.4.1.1 Apprentissage supervisé

L'apprentissage supervisé est la tâche d'apprentissage automatique consistant à apprendre une fonction qui mappe une entrée à une sortie sur la base de paires d'exemples entrée-sortie pour faire des prédictions sur des nouveaux donnés où seules les entrées sont fournies et les sorties sont prédites par le modèle, et ces prédites sont comparées aux variables cibles retenues et utilisées pour estimer la compétence du modèle. Il existe deux types principaux de problèmes d'apprentissage supervisé : la classification qui consiste à prédire une étiquette de classe et la régression qui consiste à prédire une valeur numérique[18]. Les méthodes les plus fréquemment employées dans cette catégorie sont les suivantes les réseaux neuronaux, les voisins les plus proches, les machines à vecteurs de support, les arbres de décision et l'algorithme de Bayes naïf[19].

### 1.4.1.2 Apprentissage non supervisé

L'apprentissage non supervisé consiste à n'avoir que des données d'entrée ( $X$ ) et aucune variable de sortie correspondante. Ces méthodes sont appelées "apprentissage non supervisé" parce qu'à la différence de l'apprentissage supervisé mentionné précédemment, il n'y a pas de bonnes réponses et il n'y a pas de professeur d'enseignant corrigeant le modèle, comme dans le cas de l'apprentissage supervisé. Il existe de nombreux types d'apprentissage non supervisé [19] :

- Clustering : problème d'apprentissage non supervisé qui consiste à trouver des groupes dans les données.
- Estimation de la densité : problème d'apprentissage non supervisé qui consiste à résumer la distribution des données.

### 1.4.1.3 Apprentissage par renforcement

Il explore une catégorie de situations où un agent interagit dans un environnement et doit apprendre à fonctionner en utilisant des retours obtenus grâce aux interactions avec un environnement externe. L'utilisation d'un environnement signifie qu'il n'y a pas d'ensemble de données d'entraînement fixe, mais plutôt un objectif ou un ensemble d'objectifs que l'agent doit atteindre, des actions qu'il peut effectuer, et des retours sur sa performance par rapport à l'objectif. Cela implique la poursuite d'un ou de plusieurs objectifs par l'agent, plutôt que de simplement travailler avec un ensemble de données statique [19]. Il est conscient de la boucle de rétroaction entre ses expériences et son processus d'apprentissage. Ce caractère distinctif de l'apprentissage par renforcement le rend unique par rapport aux autres méthodes d'apprentissage. On le retrouve fréquemment utilisé dans des secteurs comme la robotique, les jeux, etc .

## 1.4.2 L'apprentissage profond (DEEP LEARNING)

L'apprentissage profond (Deep learning (DL) en anglais), est un type de méthode d'apprentissage automatique. Il a été couronné de succès dans diverses applications, notamment la classification, le traitement du langage naturel, la réduction de la dimensionnalité, la détection d'objets, la modélisation de mouvement, et bien d'autres encore.

Les algorithmes d'apprentissage profond utilisent des architectures à plusieurs couches pour identifier des caractéristiques des données à différents niveaux d'abstraction, du niveau le plus bas au niveau le plus élevé, et ils peuvent découvrir d'énormes quantités de structures dans les données. Comme le processus de flux de trafic est complexe par nature, les algorithmes d'apprentissage profond peuvent représenter les caractéristiques du trafic sans connaissance préalable, ce qui donne de bonnes performances pour la prédiction du flux de trafic [20].

Pour la prédiction de flux de trafic routier, l'apprentissage profond utilise des méthodes diffé-

rentes comme les réseaux de neurones récurrents (RNN) et ces variantes la méthode Mémoire à long terme et à court terme LSTM et la méthode Unité récurrente à porte GRU.

### 1.4.3 Réseaux de neurones récurrents (RNN)

Les Réseaux de Neurones Récurrents (RNN) appartiennent à un type de modèle d'apprentissage profond. sont capables d'apprendre des caractéristiques et des dépendances à long terme à partir de données séquentielles et chronologiques, de mémoriser les données d'entrée historiques et de les utiliser pour améliorer la précision des prédictions. Les RNN sont composés d'une pile d'unités non linéaires où au moins une connexion entre les unités forme un cycle dirigé. Les RNN peuvent se souvenir des caractéristiques historiques dans les données séquentielles temporelles. Cependant, cela conduit le problème de gradient de fuite et perd le souvenir de séquences plus longues au fil du temps[2][21].

La figure 1.2 suivante représente la structure du réseaux de neurones récurrents (RNN).

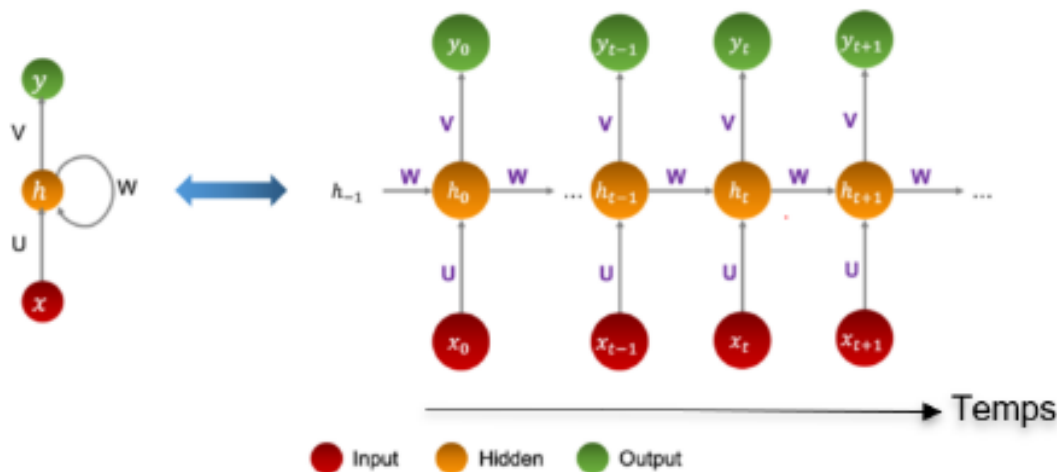


FIGURE 1.2 – Structure d'un réseaux de neurones récurrents (RNN)[2].

Deux variantes populaires de Réseaux de Neurones Récurrents (RNN) utilisées pour résoudre le problème de prédiction des flux de trafic sont les Réseaux de Mémoire à Long Terme (LSTM) et Réseaux de Neurones Récurrents Gated (GRU) ont été utilisées pour développer de nouveaux modèles de prévision des crues basés sur les données[22].

Les réseaux de mémoire à long terme (LSTM), introduits par Hochreiter et Schmidhuber en 1997 [23], sont une variante des réseaux de neurones récurrents (RNN) conçus pour apprendre des dépendances à long terme dans les données séquentielles. Contrairement aux RNN classiques, qui rencontrent des difficultés avec les problèmes de disparition et d'explosion des gradients [24], les LSTM surmontent ces obstacles grâce à une architecture interne complexe incluant des neurones

à portes (gated). Les LSTM utilisent trois portes principales : la porte d'oubli, la porte d'entrée et la porte de sortie, qui régulent le flux d'information à travers les cellules de mémoire [25]. La porte d'oubli détermine quelles informations doivent être éliminées, la porte d'entrée sélectionne les nouvelles informations à stocker, et la porte de sortie contrôle la sortie de la cellule. Les équations de fonctionnement des LSTM intègrent des fonctions d'activation non linéaires, telles que la fonction sigmoïde et la tangente hyperbolique, pour gérer les signaux de modulation. En comparaison, les unités récurrentes à porte (GRU) sont une autre variante des RNN introduite plus tard, simplifiant l'architecture LSTM en utilisant uniquement deux portes : la porte de réinitialisation et la porte de mise à jour [26], pour faciliter l'apprentissage des dépendances à long terme et surmonter les problèmes de gradient. Les GRU, tout comme les LSTM, sont conçues pour améliorer l'efficacité de la prédiction de flux de trafic en capturant à la fois les séquences à court et à long terme dans les données.

#### 1.4.4 Intégration de l'Apprentissage Automatique dans les STIs

L'intégration de l'apprentissage automatique dans les systèmes de transport intelligents (STI) représente une avancée significative pour améliorer la gestion et l'efficacité des infrastructures de transport. Cette section détaille les principaux aspects de cette intégration, y compris les applications concrètes, les études de cas, les défis et les perspectives d'avenir.

##### 1.4.4.1 Applications de l'Apprentissage Automatique dans les STI

L'apprentissage automatique offre la possibilité de traiter et d'analyser de vastes quantités de données provenant de différentes sources, comme les capteurs de trafic, les caméras de surveillance et les systèmes de positionnement GPS. Voici quelques exemples pratiques :

- **Prédiction du Trafic** : Les modèles d'apprentissage automatique, tels que les réseaux de neurones récurrents (RNN) et les réseaux de neurones convolutifs (CNN), sont employés afin de prédire en temps réel les conditions de trafic [27]. Ces modèles étudient les données passées et présentes afin de prédire les embouteillages et suggérer des alternatives de trajet .
- **Gestion du Trafic** : Les systèmes d'apprentissage automatique des STI permettent d'ajuster de manière dynamique les feux de signalisation afin d'améliorer le flux de circulation. Les schémas de circulation sont étudiés à l'aide d'algorithmes de renforcement afin d'ajuster les temps de feux en fonction [28].
- **Détection d'Incidents** : Les méthodes de vision par ordinateur, utilisant l'intelligence artificielle, permettent de repérer rapidement les accidents ou les véhicules en panne [29]. Cela

facilite l'envoi des secours de manière plus rapide et réduit les perturbations du trafic.

- **Analyse des Comportements de Conduite** : Il est possible d'analyser les données des véhicules connectés afin de saisir les comportements de conduite dangereux et suggérer des mesures de sécurité appropriées[30]. Les améliorations pour les conducteurs peuvent être suggérées par des systèmes de recommandation basés sur l'apprentissage supervisé.
- **Planification de la Mobilité Urbaine** : La planification de la mobilité urbaine repose sur l'analyse minutieuse des flux de trafic et des modèles de déplacement pour concevoir des infrastructures de transport efficaces et durables. Cette approche permet de comprendre les schémas de déplacement des usagers, d'identifier les points de congestion et de prévoir la demande future de transport. En intégrant des technologies avancées telles que les systèmes d'information géographique (SIG), le big data, et les simulations de trafic, les urbanistes peuvent visualiser et analyser les données en temps réel[30]. Ces outils aident à élaborer des stratégies pour optimiser les réseaux de transport public, synchroniser les feux de signalisation, et promouvoir des modes de déplacement durables comme le vélo et la marche.

#### 1.4.4.2 Défis de l'Intégration

- **Collecte et Traitement des Données** : la collecte et le traitement massifs des données provenant de capteurs et de dispositifs connectés nécessitent des capacités de traitement et de stockage importantes[31]. La gestion de ces énormes volumes de données exige des infrastructures informatiques robustes et évolutives. De plus, la qualité des données est essentielle pour des prédictions précises, ce qui implique des efforts continus de nettoyage et de validation des données pour garantir leur fiabilité.
- **Sécurité et Confidentialité** : la sécurité et la confidentialité des données représentent un autre défi significatif. Avec la quantité croissante de données sensibles, comme les informations de localisation des utilisateurs[32], la protection contre les cyberattaques est primordiale. Des mesures de sécurité robustes doivent être mises en place pour prévenir les violations de données et garantir la confidentialité des utilisateurs, tout en permettant l'analyse des données pour améliorer les services de transport.
- **Coût et Infrastructure** : le coût initial élevé et les besoins en infrastructure peuvent constituer un obstacle à l'adoption des STI basés sur l'apprentissage automatique, en particulier pour les régions à ressources limitées. Déployer et maintenir des infrastructures de capteurs et de traitement des données nécessite des investissements importants en argent et en ressources humaines, ce qui peut retarder leur adoption à grande échelle[33].

- **Adoption et Sensibilisation** : l'acceptation et l'utilisation efficace des STI dépendent de la sensibilisation du public et des parties prenantes. Des campagnes éducatives sont nécessaires pour informer sur les avantages des STI en termes de sécurité routière, d'efficacité des transports et d'impact environnemental[34]. Impliquer activement les citoyens, les entreprises et les décideurs locaux dans le processus peut favoriser une adoption plus rapide et une meilleure appropriation de ces technologies.

## 1.5 Conclusion

La prédiction et la détection de l'état du trafic revêt une importance capitale et a été perçue comme un défi majeur pour les systèmes de transport intelligents (ITS). Son rôle crucial dans la gestion proactive des problèmes de circulation, l'amélioration de la sécurité routière et la prestation de services de confort aux conducteurs explique son importance.

Ce chapitre a été consacré aux prérequis théoriques. Nous avons d'abord abordé les généralités de base pour notre étude, telles que l'apprentissage automatique et ses différents types, utilisés pour la prédiction de flux de trafic. Ce mémoire se concentre sur l'apprentissage supervisé, notamment sur les modèles de prédiction basés sur les réseaux de neurones récurrents (RNN), car ils sont couramment employés pour prédire les conditions de trafic. Dans le second chapitre, nous établirons un état de l'art sur quelques travaux existants dans la littérature concernant les solutions basées sur la détection et la prédiction de l'état du trafic.

# Chapitre 2

## Etat de l'art sur la prédiction et la détection de l'état du trafic

### 2.1 Introduction

Dans de nombreuses villes du monde entier, l'augmentation constante du trafic routier représente un défi majeurs en termes de gestion efficace de la circulation et de prévention des congestions, des accidents et des retards. Pour résoudre ce problème, de nombreuses études récentes ont exploré l'utilisation de l'intelligence artificielle et de l'apprentissage automatique pour améliorer l'efficacité des systèmes de transport et garantir la sécurité des usagers de la route.

Dans ce chapitre, Nous passons en revue quelques travaux existants qui examine plusieurs approches innovantes proposées dans la littérature récente pour prédire et détecte la congestion du trafic et améliorer la gestion des routes urbaines.

### 2.2 Classification des travaux passés en revus

Dans le domaine de classification de l'état de congestion, diverses approches ont été développées pour améliorer l'efficacité et la fluidité des déplacements. Ces solutions se distinguent par leur focalisation sur différents aspects de la gestion du trafic, allant de la prédiction à la détection, et parfois une combinaison des deux.

Les solutions proposées peuvent être classées en trois catégories (Figure 2.1). La première catégorie se focalise sur la détection des embouteillages grâce aux données actuelles recueillies par les capteurs . En revanche, la seconde catégorie se concentre sur la prédiction de la congestion, visant à anticiper les futurs niveaux de trafic. Enfin, la troisième catégorie adopte une approche combinée de prédiction et de détection

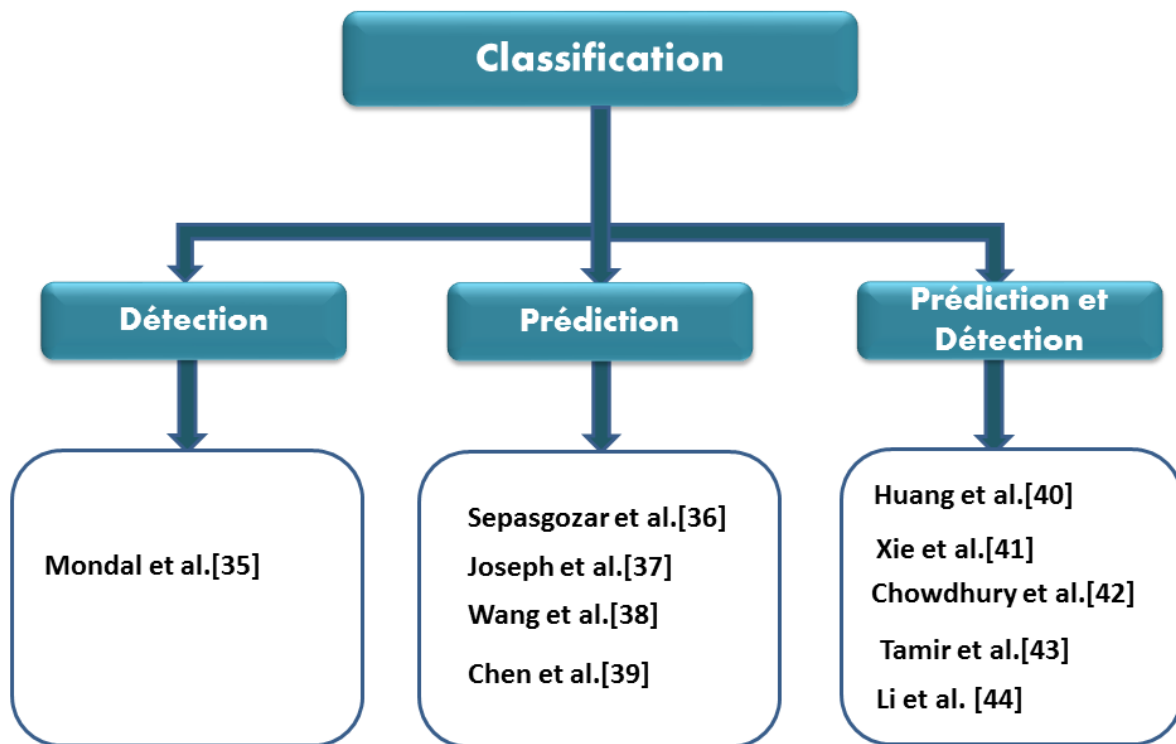


FIGURE 2.1 – Classification des travaux passés en revue.

## 2.3 Etude des travaux existants de trafic routier

### 2.3.1 Solutions basées sur la détection de l'état de trafic

Les solutions discutées ci-dessous explorent différentes approches axées sur la détection.

- **Intelligent traffic congestion classification system using artificial neural network :**

Mondal et al.[35] ont proposé un système vise à estimer et classer l'état de congestion du trafic sur différents segments de route en trois catégories : Forte Congestion, Congestion Modérée et Circulation Fluide à l'intérieur d'une ville en analysant les données de trafic routier capturées par des capteurs stationnaires en bordure de route. Le système proposé se compose de trois composants essentiels : l'unité de collecte de données, l'unité de transmission de données et l'unité de traitement de l'information. L'unité de collecte de données utilise des capteurs stationnaires en bordure de route pour surveiller le trafic en temps réel,



capturant ainsi la vitesse et le nombre de véhicules circulant sur la route. Les données collectées sont ensuite transmises de manière périodique de l'unité locale vers l'unité d'analyse à distance, soit par communication sans fil, soit par communication filaire. L'unité de traitement de l'information utilise les données brutes de trafic pour calculer la vitesse moyenne des véhicules et la densité de trafic de la route, fournissant ainsi des informations cruciales pour évaluer l'état de congestion du trafic sur différents segments de route. Le système intègre également un réseau de neurones artificiels (RNA) dans l'unité de traitement de l'information pour rendre le processus d'analyse intelligent et permettre la classification en temps réel de l'état de congestion du trafic. L'ensemble du système est conçu pour permettre une gestion efficace de la congestion du trafic.

### 2.3.2 Solutions basées sur la prédiction de l'état de trafic

Les solutions présentées ci-dessous explorent différentes approches basées sur la prédiction pour améliorer la gestion du trafic et la sécurité routière.

#### — **Network traffic prediction model considering road traffic parameters using artificial intelligence methods in VANET :**

Sepasgozar et al.[36] ont proposé un modèle pour prédire le trafic du réseau en prenant en compte les paramètres qui peuvent conduire à l'apparition du trafic routier. Le modèle proposé intègre l'algorithme de prédiction du trafic réseau Random Forest Gated Recurrent Unit Network Traffic de prédiction du trafic réseau (RF-GRU-NTP) qui prévoit simultanément le trafic réseau en fonction du trafic routier et du trafic réseau. Ce modèle comporte trois phases, la prédiction du trafic réseau basée sur la communication véhicule à unité de bord de route (V2R) tenant compte les paquets envoyés par les véhicules aux RSUs, la prédiction du trafic routier basée sur la communication de véhicule à véhicule (V2V) en tenant compte de la vitesse des véhicules, et la prédiction du trafic réseau en tenant compte de l'évolution du trafic routier sur la base des communications V2V et V2R. Le modèle hybride proposé est mis en oeuvre dans la troisième étape, pour sélectionner des caractéristiques importantes de l'ensemble de données combiné telque les communications V2V et V2R en utilisant l'algorithme d'apprentissage automatique Random Forest (RF) puis en appliquant l'algorithme d'apprentissage profond (GRU) Gating Round Robin Unit pour prédire le trafic réseau.

---

— **A novel hybrid deep learning algorithm for smart city traffic congestion predictions :**

Joseph et al.[37] traitent des accidents de la route et de la gestion du trafic dans les réseaux ad hoc véhiculaires. Le but de cette étude est de minimiser les accidents et d'améliorer la sécurité dans les réseaux ad hoc (VANET). Pour réduire le nombre d'accidents qui affectent de nombreuses personnes, il est nécessaire de prévoir les embouteillages et d'assurer une bonne gestion du trafic. Dans les réseaux ad hoc, les véhicules ont un comportement dynamique qui rend difficile l'utilisation des approches d'apprentissage profond pour réduire la congestion du trafic. Ainsi, ils ont proposé d'utiliser un réseau neuronal (CNN) pour le traitement des images et également l'utilisation de BLSTME, qui est une variante améliorée des réseaux neuronaux LSTM, permettant de traiter des séquences de données. Les auteurs de cet article ont créé une méthode combinant CNN et BLSTME, nommée BLSTME-CNN, qui permet à cette méthode d'avoir une meilleure précision dans l'analyse des données et d'identifier les caractéristiques essentielles des images. Dans le CNN, il y a deux couches importantes : la première est la couche d'extraction des caractéristiques, qui permet de trouver les éléments dans les images (les captures), et la deuxième couche s'intéresse aux informations les plus importantes. Comme BLSTME est une variante de LSTM, LSTM permet de mémoriser les informations sur les différentes périodes de l'état du trafic. Ce modèle a été implémenté en utilisant Python et la bibliothèque TensorFlow. Ils ont évalué le modèle en utilisant un scénario créé à l'aide d'un logiciel open source et du logiciel propriétaire OMNET++. Après plusieurs tests, ils ont constaté qu'il prédisait avec une précision de 98 pour cent où les embouteillages allaient se former.

— **A real-time collision prediction mechanism with deep learning for intelligent transportation system :**

Wang et al.[38] ont proposé un modèle de prédiction des collisions arrière basé sur les réseaux neuronaux convolutionnels (CNN). Ils ont utilisé une méthode de RCP (Rear-end Collision), une approche prometteuse pour améliorer la sécurité dans les villes intelligentes en prévenant les collisions arrière à l'aide d'algorithmes d'avertissement. De plus, ils ont employé des méthodes de suréchantillonnage synthétique et de prétraitement pour développer une stratégie d'avertissement efficace. Le problème qu'ils ont traité réside dans l'efficacité et la fiabilité des systèmes d'avertissement de collision arrière dans des environnements de conduite réels. Ils ont utilisé la base de données NGSIM pour étudier ce problème, qui repose sur sept entrées pour le dataset. Le traitement est basé sur le fait que les véhicules communiquent périodiquement avec les unités en bord de route (RSU) via les unités embarquées (OBU) pour collecter des informations en temps réel sur eux-mêmes et leur

environnement. Le niveau d'avertissement actuel est généré périodiquement par des réseaux neuronaux convolutifs en fonction des entrées. Le modèle CNN proposé comporte 3 couches : une couche de convolution pour le traitement des données sous forme d'images en pixels qui utilise une fonction d'activation RELU, une couche de mise en commun pour la filtration de l'information, une couche entièrement connectée, et une couche de sortie. Nous avons comparé les performances de notre modèle de prédiction proposé avec celles de Honda, Berkeley et MCMA, qui sont des algorithmes typiques de prévention des collisions arrière. Nous avons évalué la performance des algorithmes de prédiction des collisions arrière. En les utilisant, on peut évaluer à la fois la capacité d'un modèle à détecter correctement les collisions arrière (TPR) et sa capacité à minimiser les fausses alertes (FPR), ainsi que d'autres aspects de sa performance.

— **PCNN : Deep Convolutional Networks for Short-Term Traffic Congestion Prediction :**

Chen et al.[39] Dans cette étude, la méthode utilisée pour la prédiction des embouteillages à court terme repose sur un réseau de neurones convolutifs profonds (PCNN). Les caractéristiques du modèle incluent l'utilisation de convolutions multi-granulaires pour capturer les schémas de congestion à différentes échelles temporelles. Chaque convolution utilise 64 filtres de taille 2x2 pour les premières couches et 16 filtres de taille 2x2 pour la dernière couche, avec la fonction d'activation ReLU et un taux d'apprentissage fixé à 0,005. Les entrées du modèle sont organisées sous forme de matrice bidimensionnelle représentant les données de congestion périodiques. Le pliage des séries temporelles construit une matrice 2D, tenant compte des valeurs de congestion des créneaux horaires passés et des jours précédents pour capturer la périodicité et la cohérence locale du trafic. La matrice d'entrée de taille  $(d + 1) \times 2t$  permet d'inclure les conditions de trafic immédiates et les modèles historiques similaires. Les données sont complétées par des informations provenant de détecteurs de boucles inductives, de processeurs d'images vidéo et de services basés sur GPS pour une précision accrue. Les caractéristiques des entrées comprennent : Pliage des séries temporelles : Les données de congestion sont pliées temporellement pour construire une matrice 2D. Ce pliage prend en compte les valeurs de congestion des créneaux horaires passés et des jours précédents pour capturer la périodicité et la cohérence locale du trafic. Matrice d'entrée : La matrice d'entrée est de taille  $(d + 1) \times 2t$ , où  $d$  représente le nombre de jours précédents considérés et  $2t$  représente le nombre de créneaux horaires autour du créneau actuel. Cela permet d'inclure les conditions de trafic immédiates et les modèles historiques similaires. Enrichissement des données : Les données de congestion sont complétées par des données provenant de détecteurs de boucles inductives, de processeurs d'images vidéo, ainsi que des services basés sur GPS comme Google Maps pour obtenir des informations précises

sur les conditions de trafic en temps réel.

Les paramètres de performance évalués comprennent MAE, RMSE et l'erreur relative moyenne (MRE). Ces mesures permettent d'évaluer l'efficacité du modèle en comparant les valeurs prédites aux valeurs observées. Le modèle a été entraîné sur les données de circulation de 20 jours, validé sur 5 jours supplémentaires, et testé sur les 5 derniers jours pour assurer la robustesse des prédictions

### 2.3.3 Solutions basées sur la prédiction et la détection de l'état de trafic

Cette section explore des méthodes d'intelligence artificielle pour prédire et détecter les embouteillages routiers.

#### — Traffic congestion level prediction based on recurrent neural networks :

Huang et al.[40] ont proposé un système vise à estimer et classer l'état de congestion du trafic sur différents segments de route en trois catégories : Forte Congestion, Congestion Modérée et Circulation Fluide à l'intérieur d'une ville en analysant les données de trafic routier capturées par des capteurs stationnaires en bordure de route. Le système proposé se compose de trois composants essentiels : l'unité de collecte de données, l'unité de transmission de données et l'unité de traitement de l'information. L'unité de collecte de données utilise des capteurs stationnaires en bordure de route pour surveiller le trafic en temps réel, capturant ainsi la vitesse et le nombre de véhicules circulant sur la route. Les données collectées sont ensuite transmises de manière périodique de l'unité locale vers l'unité d'analyse à distance, soit par communication sans fil, soit par communication filaire. L'unité de traitement de l'information utilise les données brutes de trafic pour calculer la vitesse moyenne des véhicules et la densité de trafic de la route, fournissant ainsi des informations cruciales pour évaluer l'état de congestion du trafic sur différents segments de route. Le système intègre également un réseau de neurones artificiels (RNA) dans l'unité de traitement de l'information pour rendre le processus d'analyse intelligent et permettre la classification en temps réel de l'état de congestion du trafic. L'ensemble du système est conçu pour permettre une gestion efficace de la congestion du trafic.

#### — A data-driven lane-changing model based on deep learning :

Xie et al.[41] ont utilisé des données du programme Next Generation Simulation (NG-SIM), qui comprend des données de trajectoire de haute fidélité des véhicules sur des sections

de route de surveillance, pour développer et évaluer un modèle de comportement de changement de voie. L'article met l'accent sur modèle LC basé sur des modèles d'apprentissage profond. Le processus LC contient trois parties : préparation des données, modèle LCD (lane changing decision) et modèle LCI (lane changing implementation). le modèle LCD basé sur DBN qui est utilisé pour prédire l'LCD du conducteur introduisons une étiquette d'état pour chaque véhicule à savoir LC (lane changing) ou LK (lane keeping), avant de mettre à jour les positions des véhicules nous mettons d'abord à jour les étiquettes d'état de tous les véhicules. Si le conducteur décide de changer de voie, le modèle LCI basé sur LSTM est appliqué pour générer la trajectoire LC, qui est composé de trois couches, une couche d'entrée, une couche de sortie et une couche cachée. La couche d'entrée est conçue pour recevoir des informations sur l'état du trafic du véhicule en question et de ses véhicules environnants ( vitesses, distances entre SV et PVc, PVt, LVt et Positions longitudinales et latérale historiques ) et la couche de sortie exporte la trajectoire prédite du véhicule en question (Position longitudinale et latérale du véhicule en cours au prochain pas de temps). les résultats sont évalués à l'aide et de l'erreur quadratique moyenne (MSE).

— **A traffic congestion forecasting model using cmtf and machine learning :**

Chowdhury et al.[42] ont proposé un modèle de prévision de la congestion routière utilisant CMTF et l'apprentissage automatique. Le problème abordé dans cet article concerne les embouteillages routiers qui ont un impact grave dans de nombreuses villes à travers le monde, entraînant notamment le gaspillage de carburant et des pertes financières dues aux retards de livraison. Pour résoudre ce problème, les auteurs ont proposé un nouveau modèle de prédiction des embouteillages en utilisant la densité précalculée à partir du tableau d'informations des noeuds basé sur les données de circulation précédentes. Cette méthode est divisée en deux parties : la première consiste à effectuer une analyse de séries chronologiques sur les données précédentes d'un noeud et de ses noeuds adjacents, tandis que la deuxième permet d'effectuer des calculs basés sur ces données. Un score de 0 indique l'absence de congestion, proche de 0 signifie une faible congestion, et un score de 1 indique une circulation dense, proche de 1 signifie une voie encombrée. Le modèle CMTF utilise les noeuds adjacents à la voie que l'on souhaite évaluer pour déduire l'état de cette voie. Cette méthode exploite des cercles concentriques autour des intersections pour recueillir des données sur la congestion du trafic dans les intersections adjacentes, puis utilise ces données pour prévoir la congestion dans l'intersection cible. Cette approche permet de prendre en compte l'effet de propagation de la congestion d'une intersection à une autre. Dans notre environnement routier, une unité routière est un segment entre deux carrefours associés. Chaque unité routière est généralement bidirectionnelle et le nombre total de véhicules passant par une intersection est compté par un capteur installé à chaque traversée. Ces données sont

ensuite transmises à chaque agent routier installé sur les serveurs en bordure de route, qui calcule et prédit les embouteillages. Cette méthode ne nécessite pas de serveurs centraux ni de systèmes de sondes. Les études se sont concentrées sur deux dynamiques vitales du flux automobile : la densité des véhicules qui s'évapore de l'arrivée vers la sortie en fonction du mouvement des véhicules, et le flux dans les embouteillages qui s'évapore également de l'arrivée vers la sortie. La méthode de prévision avec le modèle CMTF comprend deux stratégies. La première concerne la prévision à long terme des embouteillages, qui consiste à réduire la congestion à partir des voies adjacentes à une heure précise. La seconde stratégie concerne la prévision à court terme des embouteillages, où l'intervalle de temps entre l'heure actuelle et l'heure de prévision est utilisé. La différence fondamentale entre ces deux stratégies réside dans le fait que la prévision à long terme ne nécessite pas d'informations sur la congestion actuelle du noeud sélectionné, contrairement à la prévision à court terme.

— **Traffic congestion prediction using decision tree, logistic regression and neural networks :**

Tamir et al.[43] ont proposé un modèle de prédiction adaptatif qui intègre parfaitement différents types de conditions de circulation pour un grand nombre de réseaux routiers. Les auteurs présentent une étude comparative des systèmes de prévision des embouteillages, notamment l'arbre de décision, la régression logistique et les réseaux de neurones. Cinq jours d'informations sur le trafic (1 231 200 échantillons) sont utilisés pour alimenter le modèle de prédiction. Les plateformes d'apprentissage automatique TensorFlow et Clementine sont utilisées pour le prétraitement des données, la formation et les tests du modèle. Premièrement, le cadre d'apprentissage automatique TensorFlow est utilisé pour modéliser le système de prédiction de congestion. Deuxièmement, l'environnement Clementine vérifie les performances du modèle de prédiction. Les plateformes d'apprentissage automatique TensorFlow et Clementine sont utilisées pour traiter les données et modéliser les algorithmes de prédiction. Le prétraitement des données commence par la catégorisation des données brutes des capteurs en différents types de structures de données, notamment les données catégorielles et les données numériques. Ils ont conçu trois types de systèmes de prévision de congestion : l'arbre de décision, la régression logistique et le réseau neuronal. L'implémentation a été réalisée en Python. L'évaluation du modèle de prédiction est basée sur l'utilisation de matrices de confusion pour mesurer les performances des algorithmes de prédiction. Les résultats montrent que l'arbre de décision a une meilleure performance de prédiction comparativement aux autres modèles. Par exemple, l'arbre de décision a obtenu une précision de 97 pour cent dans l'environnement Python et de 97,65 pour cent dans l'environnement Clementine. Les résultats indiquent que l'arbre de décision est le modèle le plus performant parmi les trois modèles de prédiction étudiés.

— **Short-term traffic congestion prediction with Conv-BiLSTM considering spatio-temporal features :**

Li et al. [44] ont développé un modèle de détection et prédiction de la congestion du trafic qui combine des réseaux de neurones convolutifs (CNN) et des mémoires à long terme bidirectionnelles (BiLSTM), en intégrant des caractéristiques spatio-temporelles des données de trafic. Les données de vitesse du trafic, organisées en une matrice tridimensionnelle, constituent les principales entrées du modèle. Cette matrice capture les variations de la vitesse du trafic à travers différentes périodes récentes et des schémas historiques similaires, permettant une modélisation plus précise des conditions de trafic. L'indice de congestion est basé sur la vitesse et calcule comme

$$c = \frac{sf}{F5(sr)}$$

ou (sf) est la vitesse de référence en conditions idéales et (sr) la vitesse réelle, offrant une mesure intuitive de l'état de la route.

Le modèle utilise le CNN pour extraire les caractéristiques spatiales de la matrice de trafic, tandis que le BiLSTM capture les caractéristiques temporelles, permettant ainsi de comprendre les dépendances à long terme dans les données. Les résultats des expérimentations menées sur les données de trafic d'une autoroute à Shanghai montrent que le modèle Conv-BiLSTM offre des prédictions de congestion plus précises que les méthodes traditionnelles. En termes de performance, le modèle présente une amélioration notable des métriques d'évaluation comme le RMSE (Root Mean Square Error) et le MAE (Mean Absolute Error), indiquant une précision accrue dans les prédictions. Par exemple, l'article rapporte que le Conv-BiLSTM réduit significativement le RMSE et le MAE par rapport à des modèles comme ARIMA et LSTM classiques.

Les contributions majeures de l'article incluent l'utilisation innovante d'une matrice tridimensionnelle pour modéliser les données de trafic, l'introduction d'un indice de congestion basé sur la vitesse pour une visualisation intuitive, et la démonstration que le modèle Conv-BiLSTM améliore significativement les performances de prédiction par rapport aux approches existantes. Les expérimentations confirment que ce modèle est capable de fournir des prédictions fiables et précises, facilitant ainsi une meilleure gestion et planification du trafic urbain. Les futures recherches pourraient se concentrer sur l'intégration de données supplémentaires et l'amélioration de l'algorithme pour gérer les variations de trafic imprévues.

## 2.4 Etude comparative

### 2.4.1 Critères de comparaison

Pour évaluer et comparer les différents travaux étudiés, nous avons établi une série de critères. Ces critères de comparaison ont été soigneusement sélectionnés en fonction de leur pertinence pour la gestion du trafic routier. Les principaux critères retenus sont les suivants :

1. **Méthode** : est une approche ou un ensemble de techniques utilisées pour analyser, prévoir et surveiller les conditions de trafic routier, il existe des différentes méthodes comme : ANN, CNN, LSTM, etc.
2. **Caractéristique** : Les types de données qui sont utilisées pour entraîner les modèles, telles que la vitesse, la densité, l'identifiant du véhicule, les images, etc.
3. **Simulateur** : l'environnement de simulation utilisé pour implémenter et tester les modèles (Python, TensorFlow).
4. **Coût** : Pour détecter et prédire le trafic routier de manière rapide et efficace, il est nécessaire d'utiliser des technologies avancées. Cependant, ces technologies peuvent être coûteuses en termes de temps et de complexité de calcul.
5. **Paramètre de performance** : Les paramètres de performance sont les indicateurs et mesures utilisés pour évaluer l'efficacité et la précision des méthodes ou algorithmes appliqués à la prédiction et la détection du trafic des métriques différentes :

- **Erreur Quadratique Moyenne (MSE)** : Elle fournit une indication de la dispersion des erreurs et de la qualité globale de la prédiction. Plus l'erreur quadratique moyenne est faible, plus les prédictions du modèle sont proches des valeurs réelles[45].
- **Erreur Absolue Moyenne (MAE)** : Elle donne une indication directe de la magnitude moyenne des erreurs de prédiction[45].
- **Racine de l'Erreur Quadratique Moyenne (RMSE)** : Elle offre une interprétation plus intuitive de l'écart moyen entre les valeurs prédites et les valeurs réelles[45].
- **Précision (Accuracy)** : utile pour évaluer les performances des modèles de classification. Une précision élevée indique que le modèle fait un bon travail pour classer correctement les données[46].
- **Rappel (Recall)** : Le rappel évalue la capacité du modèle à détecter correctement tous les cas positifs présents dans les données[46].



## 2.4.2 Tableau comparatif

Le tableau 2.1 illustre une étude comparative des travaux passés en revues.

Classification	Travaux	Méthode	Caractéristique	Coût	Paramètre de performance
Détection	[35]	ANN	Densité, vitesse moyenne	Élevé	-
Prédiction	[36]	RF-GRU-NTP	Vitesse	Moyen	MAE=1.437, MSE=0.372, RMSE=1.199
	[37] [38]	CNN-BLSTM RCP,CNN	Image Identifiant, vitesse, accélération	Élevé Élevé	- Rappel=94%, Precision=96%
	[39]	PCNN	processeurs d'images vidéo et des services basés sur GPS	Moyen	MRE=74% , MAE=0.05
Prédiction et Détection	[40]	LSTM-RNN	Vitesse, densité	Élevé	-
	[41]	LSTM-DBN	Vitesse, densité	Élevé	-
	[42]	CMTF	Densité	Moyen	-
	[43]	Apprentissage automatique, arbre de décision	Vitesse, densité, type, ID-lieu	Moyen	Precision=95%, Rappel=96%
	[44]	Conv-BiLSTM	vitesse	Moyen	RMSE=0.849 MAE=0.388

TABLE 2.1 – Tableau comparatif.

### 2.4.3 Discussion

L'analyse des travaux passés révèle que certaines études se concentrent exclusivement sur la détection ou la prédiction, tandis que d'autres abordent ces deux aspects simultanément en utilisant des méthodes d'intelligence artificielle adaptées aux caractéristiques spécifiques de chaque approche. Les performances de ces modèles sont évaluées selon divers paramètres pour déterminer leur efficacité.

Les systèmes basés sur la détection, comme celui proposé par Mondal et al.[35], utilisent des réseaux de neurones artificiels (ANN) pour surveiller en temps réel la vitesse et la densité du trafic à l'aide de capteurs stationnaires. Bien que cette approche permette une gestion efficace de la congestion en fournissant des informations précises et en temps réel, elle est coûteuse en termes de ressources computationnelles et ne permet pas d'anticiper les événements futurs. Les modèles de détection sont limités à l'analyse des données actuelles, ce qui peut restreindre leur capacité à fournir des informations exploitables pour une gestion proactive du trafic.

Les méthodes de prédiction, comme le modèle RF-GRU-NTP de Sepasgozar et al.[36], exploitent les techniques d'intelligence artificielle pour anticiper les conditions de trafic futures. En combinant des algorithmes tels que Random Forest pour la sélection de caractéristiques et GRU pour la modélisation des séries temporelles, ce modèle offre une précision accrue dans la prédiction du trafic. De même, l'approche hybride CNN-BLSTM de Joseph et al.[38] utilise des réseaux neuronaux convolutifs pour extraire les caractéristiques spatiales des données de trafic et des réseaux BLSTM pour capturer les dépendances temporelles, offrant ainsi une précision améliorée dans la prédiction des embouteillages. Cependant, les modèles de prédiction peuvent manquer de réactivité face aux événements en temps réel, limitant leur capacité à ajuster rapidement les prévisions en fonction des nouvelles données. ce qui pourrait entraîner des prédictions moins précises et une capacité de généralisation réduite du modèle.

Parmi les modèles examinés, le ConvBiLSTM[44] émerge comme une approche prometteuse pour la détection et la prédiction du trafic. En combinant des couches convolutives avec des cellules LSTM bidirectionnelles, ce modèle est capable de capturer à la fois les informations spatiales et temporelles dans les données de trafic. Des travaux antérieurs ont démontré que le ConvBiLSTM peut offrir des performances supérieures en termes de précision et de réactivité par rapport à d'autres architectures, comme le modèle LSTM simple. Cependant, le modèle ConvBiLSTM[44] peut être limité par le nombre d'entrées utilisées pour l'entraînement, qui dans certains cas se limite à des données de vitesse uniquement. Cela peut poser un défi significatif, car les prédictions de trafic reposent souvent sur une variété de données, telles que la densité du trafic, les conditions météorologiques, et d'autres facteurs. Par conséquent, l'utilisation exclusive de la vitesse comme entrée peut ne pas être suffisante pour capturer toutes les nuances et les variations du trafic, ce qui pourrait entraîner des prédictions moins précises et une capacité de généralisation réduite du modèle.

En conclusion, pour surpasser les limitations inhérentes aux modèles de prédiction et de détection pris individuellement, il est impératif d'intégrer ces deux capacités. Ainsi, nous envisageons de concevoir un modèle hybride qui fusionne de manière efficace ces deux approches. Ces modèles hybrides sont capables de traiter les données spatio-temporelles en identifiant les motifs locaux tout en capturant les dépendances temporelles.

De surcroît, afin d'accroître les performances du ConvBiLSTM, nous avons envisagé d'incorporer des entrées supplémentaires. En enrichissant les données d'entrée avec d'autres caractéristiques pertinentes, telles que la densité du trafic ou les conditions météorologiques, nous aspirons à renforcer la capacité du modèle à appréhender la complexité du trafic routier et à améliorer la précision de ses prédictions.

## 2.5 Conclusion

Dans ce chapitre, nous avons exploré divers travaux de recherche visant à améliorer la gestion du trafic routier en utilisant des techniques d'intelligence artificielle. Nous avons examiné des approches de prédiction, de détection, ainsi que des méthodes hybrides combinant les deux. Ces études ont mis en évidence l'importance croissante de l'IA dans la résolution des défis liés à la congestion et à la sécurité routière.

Dans le chapitre suivant, nous présenterons un modèle de prédiction et de détection de l'état du trafic visant à améliorer la précision des prédictions et des détections. Nous présenterons la conception générale et détaillée de ce système proposé. Cette nouvelle approche repose sur la prédiction et la détection de l'état du trafic à l'aide d'une méthode d'apprentissage profond, étudiée pour optimiser les performances du modèle et fournir des prédictions et des détections plus fiables et précises.

# Chapitre 3

## Systeme de Prédiction et Détection de congestion (Bi-ConvLSTM)

### 3.1 Introduction

Dans les chapitres précédents, nous avons présenté la partie théorique qui explique les notions de base conceptuelle essentielle de notre projet et quelques travaux précédents qui nous ont permis de proposer un modèle de prediction et detection de congestion du trafic. Notre approche centrée sur l'utilisation du deep learning, spécifiquement des réseaux neuronaux, pour définir l'état du trafic.

Dans ce chapitre, nous allons aborder la conception essentielle de notre travail : la mise en place du système proposé appelé Bi-ConvLSTM. Tout d'abord, nous présenterons l'architecture générale du système proposé (Bi-ConvLSTM) de prédiction et de détection de flux de trafic, en mettant l'accent sur l'hyperparamétrage. Ensuite, nous décrirons les différentes étapes de cette architecture en détail, en explicitant le fonctionnement de chacune de ces étapes.

### 3.2 Motivation

La congestion du trafic est un phénomène global étendu pour les systèmes de transport urbain, engendrant des retards, des inefficacités économiques et des dommages environnementaux considérables. Devant cette situation, la sécurité routière émerge comme l'une des préoccupations les plus cruciales à l'échelle mondiale. La prédiction et la détection de la congestion du trafic joue ainsi un rôle essentiel dans le développement et l'amélioration des systèmes de transport intelligents. En anticipant les zones de congestion et en fournissant des informations en temps réel sur les conditions de circulation, les autorités peuvent mettre en place des mesures préventives pour réduire les risques d'accidents et optimiser l'efficacité des déplacements.

C'est dans ce contexte que notre travail prend tout son sens. Nous sommes particulièrement

motivés par le modèle ConvLSTM-BiLSTM [44], qui a démontré des résultats prometteurs dans ce domaine. Ce modèle combine une mémoire convolutive à long terme (ConvLSTM) pour capturer les caractéristiques spatio-temporelles du trafic et une mémoire bidirectionnelle à long terme (BiLSTM) pour extraire les caractéristiques périodiques des flux de trafic.

En effet, le modèle ConvLSTM-BiLSTM de Li et al.[44] utilise principalement la vitesse comme entrée pour juger de l'état du trafic. Or, la congestion ne dépend pas uniquement de la vitesse. D'autres facteurs, tels que le volume du trafic, la densité des véhicules, jouent un rôle crucial dans la formation des embouteillages. Ainsi, pour améliorer la précision de notre modèle, nous prévoyons d'améliorer encore la précision de nos prédictions en ajoutant plusieurs paramètres et entrées spécifiques.

En fournissant aux gestionnaires de transport des informations exploitables, notre travail vise à réduire les risques d'accidents, à minimiser les retards et à optimiser les flux de circulation dans les environnements urbains.

### 3.3 Conception générale du BiConvLSTM

La prédiction et détection de la congestion du trafic consiste à anticiper les endroits où la circulation routière pourrait devenir très dense, voire se bloquer complètement. Notre approche pour y parvenir consiste à utiliser différents types d'informations sur le trafic, comme la densité de véhicules, la vitesse et la distance entre les véhicules. En regroupant ces données et en les analysant, notre système permet de détecter l'état de trafic en temps réel et de prédire la congestion routière. Cela permet aux autorités de prendre des mesures préventives pour éviter les embouteillages et garantir la sécurité sur les routes. L'architecture de notre système de prédiction et détection de la congestion du trafic est illustrée dans la Figure 3.1. Comme le montre cette architecture, notre système peut être divisé en six étapes :

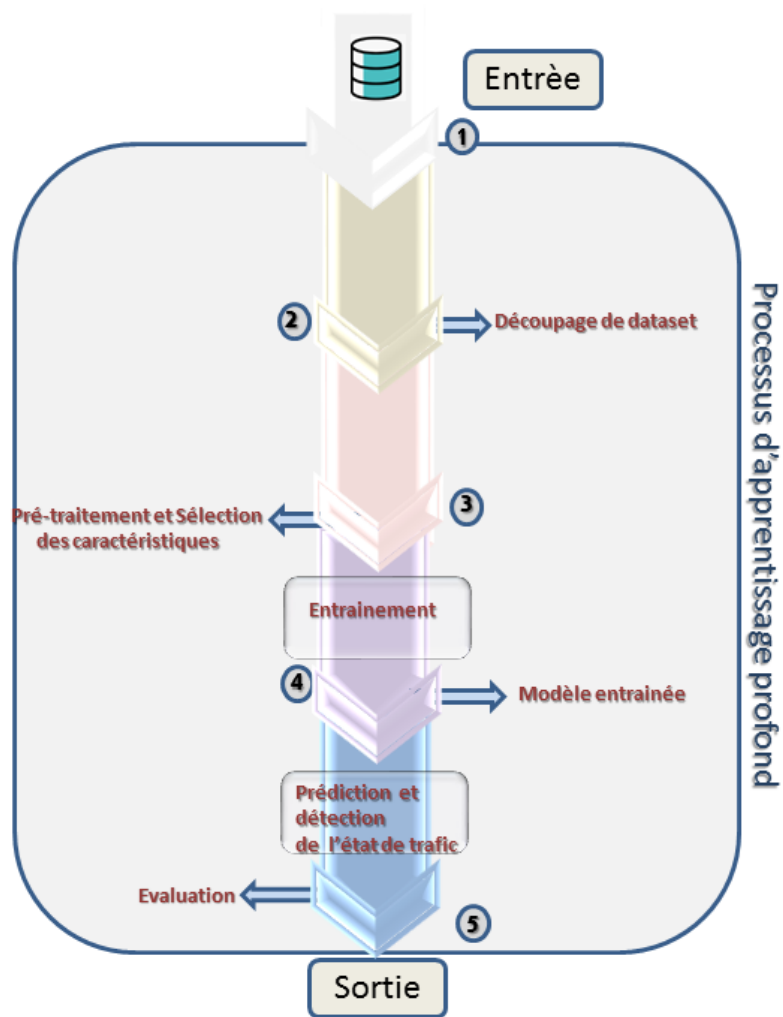


FIGURE 3.1 – L'architecture générale du Bi-ConvLSTM.

### 3.4 Conception détaillée du Bi-ConvLSTM

Dans cette section, nous allons exposer et décrire en détail les différentes étapes que notre système va suivre, comme illustré dans la Figure 3.2.

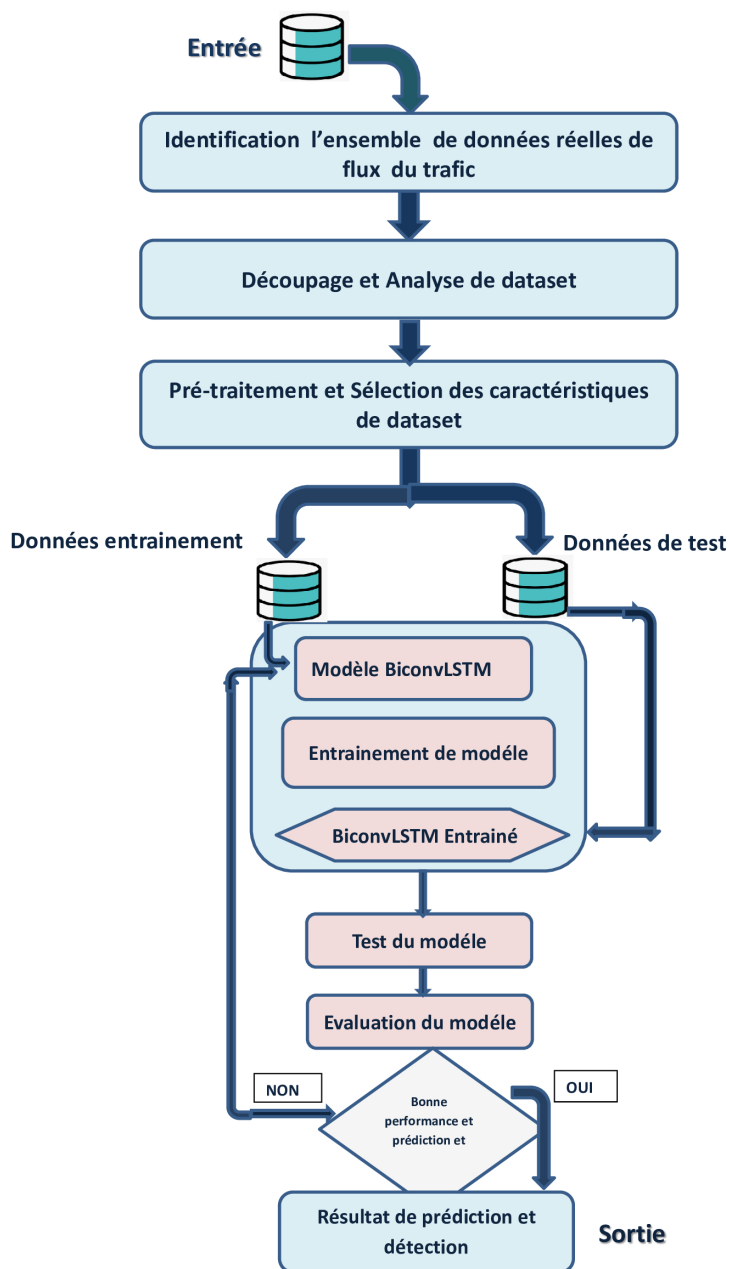


FIGURE 3.2 – Architecture détaillée du Bi-ConvLSTM.

### 3.4.1 Identification de l'ensemble de données

L'ensemble de données dit aussi Dataset, c'est une série temporelle est constituée d'observations numériques ordonnées chronologiquement représentant une diversité d'éléments. Le choix de dataset (jeu de données) est un élément important dans la réussite de la mise en place d'un modèle d'apprentissage profond, car il définit le jeu de données sur lequel le modèle sera entraîné.

Dans le cadre de notre projet, l'évaluation des mesures de congestion a été réalisée à l'aide d'un ensemble de données open source provenant du suivi du trafic de Chicago [47]. Cet ensemble de

données se compose des données historiques de congestion estimée pour plus de 1 000 segments de trafic, à partir d'environ mars 2018. La congestion du trafic sur les rues artérielles de Chicago (rues non autoroutières) en temps réel. Cet ensemble de données contient des données sur les trajets de véhicule, incluant la vitesse (SPEED), la longueur du segment (LENGTH) pour chaque intervalle de 10 minutes pendant sept jours consécutifs. le nombre de véhicules, et la distance moyenne entre les véhicules (Distance Moyenne).

Pour notre travail, nous avons pris en compte une journée entière de données du 26 août 2023, pour un segment allant de Canal à Halsted. La vitesse, mesurée en miles par heure (mph), varie considérablement. La longueur du segment est constante à 0,4 miles pour toutes les entrées. Tandis que la vitesse fluctue en fonction de divers facteurs tels que le trafic ou les arrêts fréquents. Il existe une relation inverse entre le nombre de véhicules et la distance moyenne entre les véhicules : plus le nombre de véhicule sur un segment est élevé, plus la distance moyenne entre eux est réduite. Le dataset utilisé dans notre système est une série temporelle qui contient une suite d'observations chiffrées ordonnées dans le temps, et considéré comme une collection d'éléments variés. En outre, notre dataset est sous forme de Fichier.csv utilisé comme entrée dans le processus d'apprentissage profond. La figure 3.3 suivante représente la zone d'étude dans Chicago.



FIGURE 3.3 – Le segment d'étude.



La description des champs de données est présentée dans tableau 3.1 :

Nom de champs	Description
Vitesse (Miles)	La vitesse du trafic, ou vélocité, $v$ est définie comme la vitesse estimée du trafic en miles par heure : une valeur de -1 signifie aucune estimation.
longueure(miles)	Longueur du segment en miles
Densité	Nombre de vehicules fournissant un flux GPS utilisé pour estimer la congestion routière.
Distance Moyenne(Miles)	est calculée en divisant la longueur par le nombre de vehicules.

TABLE 3.1 – la description des champs de dataset.

### 3.4.2 Découpage et analyse de dataset réelles

Dans cette partie, nous aborderons la manière dont nous avons découpé et analysé le dataset réel afin de le préparer pour l’entraînement de notre modèle de prédiction et de détection de la congestion du trafic.

#### 3.4.2.1 Découpage de dataset réelles

Dans la première étape nous découpons notre dataset en trois parties : entraînement et validation et test, Chaque partie a une signification importante.

1. **Données d’entraînement** : Ces données sont utilisées pour l’apprentissage d’un modèle, elles servent à entraîner le modèle afin qu’il puisse apprendre à partir des exemples fournis.
2. **Données de validation** : Il s’agit d’un ensemble de données utilisé pour évaluer de manière impartiale la performance d’un modèle pendant le processus d’entraînement. Ces données permettent également d’ajuster les hyperparamètres du modèle afin d’optimiser ses performances.
3. **Données de test** : Ces données sont utilisées pour évaluer la performance du modèle final et vérifier les résultats obtenus. Elles fournissent une évaluation indépendante de la capacité du modèle à généraliser à de nouvelles données.

Dans notre étude, nous suivons la stratégie la plus courante en apprentissage profond, qui consiste à diviser le dataset en ensembles d’entraînement, de validation et de test. Le rapport optimal d’échantillons répartis dans chaque ensemble varie en fonction du problème spécifique.

Dans notre cas, nous avons séparé le dataset en un ensemble d'entraînement et un ensemble de test. La majorité des données sont utilisées pour l'entraînement (90%), tandis qu'une plus petite partie est réservée au test (10%). De plus, nous réservons également une portion des données d'entraînement (20%) pour la validation, ce qui nous permet d'évaluer le modèle.

### 3.4.2.2 Analyse de dataset réelles

Analyse de dataset est une étape la plus importante de notre processus qui nous permet de nous donner une idée de ce que contient l'ensemble de données afin de le préparer pour le rendre exploitable par le modèle, et aussi d'étudier les données afin de comprendre les particularités de la série. La série contient différents champs telles que : Type de données, Nombre d'observations, Le nombre de features, Les variables connues, Les variables à prédire, Détecter les valeurs manquantes, Détecter des valeurs Null et Détecter des valeurs qui ne sont pas numériques. Ceci nous permet de sélectionner les étapes adéquates pour prétraiter et nettoyer les données.

Grâce à cette méthode de division, il est possible de réaliser différents entraînements en utilisant le même ensemble de données d'entraînement pour rechercher les hyperparamètres optimaux afin de maximiser les performances sur l'ensemble de validation. Une fois que les meilleures performances sont obtenues sur l'ensemble de validation, le modèle est finalement utilisé sur l'ensemble de test pour mesurer et confirmer ses performances finales. Après analyse, nous constatons que ce jeu de données ne contient pas de valeurs manquantes ou nulles. Par conséquent, nous procéderons à un prétraitement afin de préparer nos données pour l'entraînement et l'évaluation du modèle.

## 3.4.3 Pré-traitement et sélection des caractéristiques

### 3.4.3.1 Pré-traitement de dataset

Dans notre processus de prétraitement des données, nous menons trois opérations distinctes pour améliorer la qualité et la performance des modèles prédictifs :

- **Nettoyage des données** : Dans notre travail, lors de l'analyse de l'ensemble de données, nous avons identifié la présence de valeurs négatives, ainsi que la valeur spécifique de -1 dans la colonne de vitesse, indiquant qu'aucune estimation n'est disponible à ce moment-là. Dans certains cas, une approche directe consiste à supprimer ces valeurs négatives pour garantir que notre ensemble de données est propre, fiable et prêt à être utilisé pour l'analyse ou la modélisation.

- **La normalisation** : Est réalisée à l'aide de la classe `MinMaxScaler` de la bibliothèque `sklearn.preprocessing`. Le processus commence par l'initialisation du scaler. Ensuite, le scaler est ajusté sur les données d'entraînement en utilisant la méthode `fit_transform()`,

ce qui permet au scaler d'apprendre les valeurs minimales et maximales des données d'entraînement, puis de transformer ces données pour qu'elles se situent entre 0 et 1. Les données de validation et de test sont ensuite transformées en utilisant le même scaler avec la méthode `transform()`, assurant ainsi la cohérence des transformations sur tous les ensembles de données. La formule utilisée pour la normalisation des données est [48] :

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}}$$

où  $x$  est la valeur d'une caractéristique avant normalisation.

$x_{\min}$  est la valeur minimale de cette caractéristique dans les données d'entraînement.

$x_{\max}$  est la valeur maximale de cette caractéristique dans les données d'entraînement.

$x'$  est la valeur normalisée de la caractéristique.

#### • Conversion des données de séries temporelles multivariées aux séries temporelles supervisées :

La gestion des séries temporelles multivariées présente des défis uniques, notamment en raison de la dépendance complexe entre différentes variables au fil du temps. Les méthodes traditionnelles d'analyse des séries temporelles peuvent souvent être insuffisantes pour capturer ces relations complexes, ce qui entraîne des difficultés de traitement et de modélisation. Pour surmonter ces obstacles, une approche couramment utilisée consiste à transformer la série temporelle multivariée en une série temporelle supervisée, où chaque observation est associée à des valeurs passées de toutes les variables ainsi qu'à la valeur future à prédire. La méthode de la fenêtre glissante, également appelée "sliding window", est souvent utilisée pour effectuer cette transformation. En utilisant cette technique, les données sont remodelées pour être compatibles avec les modèles d'apprentissage profond, tels que les réseaux de neurones récurrents (RNN) ou les réseaux de neurones convolutifs (CNN), facilitant ainsi la prédiction précise des séries temporelles multivariées.

#### 3.4.3.2 Sélection des caractéristiques

Une fois les données prétraitées, nous avons obtenu un ensemble de données (`expox.csv`) prêt pour l'entraînement et le test de notre modèle de prédiction et détection de la congestion du trafic. Pour chaque expérience, nous avons mené une sélection rigoureuse des caractéristiques à inclure en tant que variables d'entrée pour notre modèle, en nous appuyant sur les exigences spécifiques de notre étude. En examinant attentivement les données et en considérant l'objectif de notre modèle de déterminer l'état de la route, nous avons choisi différentes caractéristiques telles que :

- **Vitesse moyenne du segment** : La vitesse moyenne des véhicules sur un segment de route à chaque instant de temps.

- **Densité** : Le nombre de véhicules par unité de longueur sur le segment de route.
- **Distance moyenne** : La distance moyenne entre les véhicules sur le segment de route, que nous avons calculée. La formule utilisée est la suivante [49] :

$$\text{Distance moyenne} = \frac{\text{Longueur du segment}}{\text{Nombre de véhicules}} \quad (3.1)$$

Afin d'évaluer les conditions actuelles de circulation par rapport à des conditions optimales, nous avons effectué des calculs pour déterminer la vitesse optimale, la densité optimale et la distance moyenne optimale. Ces métriques permettent de déterminer l'état de la route, que nous classifions en trois catégories : congestionné, fluide et moyennement congestionné.

### 1. Calcul de l'Indice de Vitesse :

L'indice de vitesse est un ratio qui compare la vitesse actuelle d'un véhicule à la vitesse maximale autorisée sur un segment de route. Cet indice permet d'évaluer dans quelle mesure la vitesse d'un véhicule respecte la limite imposée et aide à classer l'état du trafic.

La vitesse indiquée pour chaque véhicule  $v_{\text{indice}_i}$  est calculée comme suit [50] :

$$v_{\text{indice}_i} = \frac{v_i}{\text{Maximum\_allowed\_speed\_current\_road}} \quad (3.2)$$

où :

- $v_i$  est la vitesse du véhicule  $i$ .
- `Maximum_allowed_speed_current_road` est la vitesse maximale autorisée sur le segment de route.

### 2. Calcul de la Densité Maximale :

la densité maximale est le nombre maximum de véhicules qui peuvent physiquement être présents sur ce segment, compte tenu de la longueur typique d'un véhicule et du nombre de voies disponibles. Cette métrique est importante pour comprendre le potentiel de saturation du trafic sur une route donnée [51].

$$\rho_{\max} = \frac{L_{\text{vehicule}}}{L_{\text{segment}}} \times N_{\text{voies}} \quad (3.3)$$

où :

- $\rho_{\max}$  est la densité maximale de véhicules.
- $L_{\text{vehicule}}$  est la longueur d'un véhicule.
- $L_{\text{segment}}$  est la longueur du segment de route.
- $N_{\text{voies}}$  est le nombre de voies sur le segment de route.

### 3. Calcul de la distance de sécurité :

Nous avons élaboré une fonction sophistiquée pour calculer la distance de sécurité entre les véhicules, prenant en considération précisément la vitesse actuelle du véhicule et le temps de réaction moyen du conducteur[52] .

$$d_{\text{securite}} = \frac{\text{Vitesse}}{3.6} \times \text{Temps de réaction} \quad (3.4)$$

où :

- Vitesse : est la vitesse du véhicule en mètres par seconde (m/s)
- Temps de réaction : est le temps de réaction du conducteur en secondes.

### 4. Conditions et Catégories pour l'État de la Route :

Par la suite, nous avons utilisé ces valeurs optimales afin de classer l'état de la route en trois catégories : congestion, moyenne, fluide. Afin d'accomplir cette tâche, nous avons établi des limites pour chaque paramètre et avons comparé les valeurs des données à celles optimales :

- **Catégories de Vitesse** : Les catégories de vitesse classent les véhicules en fonction de leur vitesse par rapport à des seuils prédéfinis, permettant ainsi d'évaluer leur comportement sur la route. Les formules utilisées sont les suivantes :

$$\text{vitesse}_{\text{catégorie}} = \begin{cases} \text{Rapide} & \text{si } v_i > v_1 \\ \text{Moyenne} & \text{si } v_1 \leq v_i \leq v_2 \\ \text{Faible} & \text{sinon } v_i < v_2 \end{cases}$$

- **Catégories de densité** : Les catégories de densité sur une route classifient le niveau de trafic en fonction du nombre de véhicules présents. Une densité élevée indique une congestion sévère, avec une capacité routière dépassée à plus de 80%. Une densité moyenne représente un trafic modéré, utilisant entre 50% et 80% de la capacité maximale. Une densité faible signifie que la route est peu encombrée, avec moins de 50% de la capacité maximale utilisée, assurant une circulation fluide et efficace. Les formules sont les suivantes :

$$\rho_{\text{catégorie}} = \begin{cases} \text{Élevée} & \text{si } \rho > 0.8 \rho_{\text{max}} \\ \text{Moyenne} & \text{si } 0.5 \rho_{\text{max}} \leq \rho \leq 0.8 \rho_{\text{max}} \\ \text{Faible} & \text{sinon } \rho < 0.5 \rho_{\text{max}} \end{cases}$$

- **Catégories de Distance de Sécurité** : Enfin, on peut considérer les différentes catégories de distance de sécurité basées sur la proximité entre les véhicules, ce qui est crucial pour évaluer le risque de collision potentielle en fonction de la vitesse du véhicule et du temps de réaction du conducteur.

$$\bar{d}_{\text{catégorie}} = \begin{cases} \text{Proximité dangereuse} & \text{si } d < 0.5 d_{\text{sécurité}} \\ \text{Distance modérée} & \text{si } 0.5 d_{\text{sécurité}} \leq d < d_{\text{sécurité}} \\ \text{Distance de sécurité respectée} & \text{sinon } d \geq d_{\text{sécurité}} \end{cases}$$

où :  $d_{\text{sécurité}}$  est la distance de sécurité recommandée calculée.

- **Distance de Sécurité respecter** : Cette catégorie indique que la distance actuelle ( $d$ ) entre votre véhicule et celui qui vous précède est suffisante pour permettre un freinage sûr en cas de besoin.
- **Proximité Dangereuse** : Cette catégorie signifie que la distance actuelle ( $d$ ) est insuffisante pour permettre un freinage efficace en cas de besoin.

- **Classification et entraînement du modèle** : Le trafic est classifié en trois catégories en fonction des conditions ci-dessus :

$$\text{trafic}_{\text{catégorie}} = \begin{cases} \text{Congestionné si } (v_i < v_2) \wedge (\rho > 0.8 \rho_{\text{max}}) \wedge (d < 0.5 d_{\text{sécurité}}) \\ \text{Moyenne si } (v_1 \leq v_i \leq v_2) \wedge (0.5 \rho_{\text{max}} \leq \rho \leq 0.8 \rho_{\text{max}}) \wedge (0.5 d_{\text{séc}} \leq d < d_{\text{séc}}) \\ \text{Fluide sinon } (v_i > v_1) \wedge (\rho < 0.5 \rho_{\text{max}}) \wedge (d \geq d_{\text{sécurité}}) \end{cases}$$

Ces conditions permettent de déterminer l'état du trafic en fonction des paramètres mesurés.

### 3.4.4 Modélisation et entraînement du modèle

Nous avons opté pour une méthode d'apprentissage profond issue de l'intelligence artificielle, basée sur un réseau de neurones récurrents (RNN). Le modèle, appelé Encodeur-Décodeur BiConvLSTM, intègre à la fois des techniques de détection et de prédiction dans l'analyse des données de trafic. Ce modèle (figure 3.4) est composé d'une mémoire convolutive à long terme (ConvLSTM2D) comme un encodeur couplée avec une LSTM bidirectionnelle (Bi-LSTM). Le ConvLSTM2D intègre des opérations de convolution directement dans les unités LSTM pour traiter les entrées, permettant ainsi d'extraire

des informations spatio-temporelles bidimensionnelles à partir des données de flux de trafic. Pour la détection, le modèle utilise la mémoire convolutive à long terme (ConvLSTM2D), ce qui lui permet d'analyser les schémas historiques de flux de trafic et de détecter la fréquence du trafic pour des prédictions plus précises. En ce qui concerne la prédiction, le modèle utilise une LSTM bidirectionnelle (Bi-LSTM) qui prend en compte à la fois les données du passé et du futur à un moment précis. Cela lui permet d'étudier les données historiques de flux de trafic, ce qui permet de détecter la fréquence du flux de trafic afin de faire des prédictions plus précises.

(figure3.4)

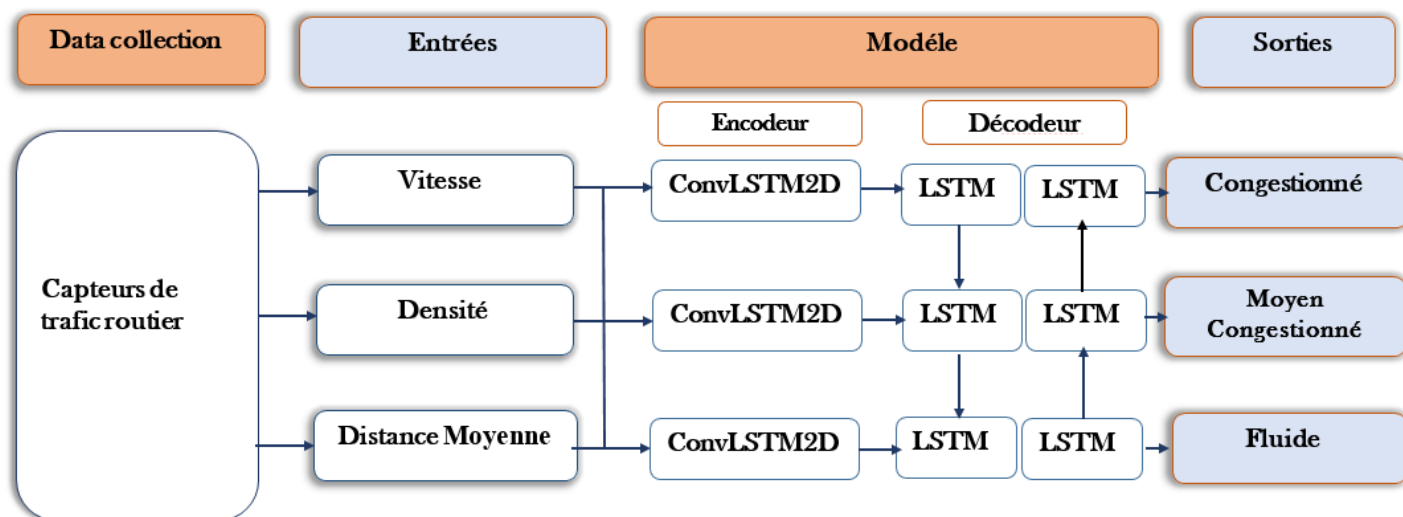


FIGURE 3.4 – Modèle Bi-ConvLstm .

Le modèle présenté ci-dessus (la figure 3.5) se compose de plusieurs principaux, chacun jouant un rôle essentiel dans le traitement et l'analyse des données pour des prédictions précises.

#### • Les couches ConvLSTM2D

Deux couches ConvLSTM2D avec 64 filtres et une taille de noyau de (1, 1). Elles extraient les caractéristiques spatiales et temporelles des données d'entrée en appliquant des opérations de convolution tout en maintenant une mémoire à court et long terme pour capturer les dépendances temporelles.

#### • Les couches Dropout

Nous utilisons les couches Dropout dans le but de désactiver aléatoirement une fraction des neurones pendant l'entraînement pour prévenir le surapprentissage.

#### • Couche Flatten

Nous ajoutons une couche Flatten qui permet de transformer les sorties des couches précédentes en un format unidimensionnel, permettant ainsi le passage fluide des données aux couches suivantes.

•**Couche RepeatVector**

Cette couche ajuste la séquence temporelle en répétant la séquence de vecteurs pour les reproduire par les couches encodeurs.

•**Couche LSTM bidirectionnelle**

Une couche LSTM bidirectionnelle avec 256 unités reprend la séquence des vecteurs de la couche de répétition et génère une prédiction sous la forme d'une séquence de vecteurs à une seule ligne, capturant les dépendances temporelles bidirectionnelles.

•**Couches TimeDistributed(Dense)**

Une couche TimeDistributed(Dense) avec 50 unités applique une couche Dense (complètement connectée) à chaque pas de temps de l'entrée. Une seconde couche TimeDistributed(Dense) avec 3 unités de sortie est utilisée pour produire les prédictions finales.



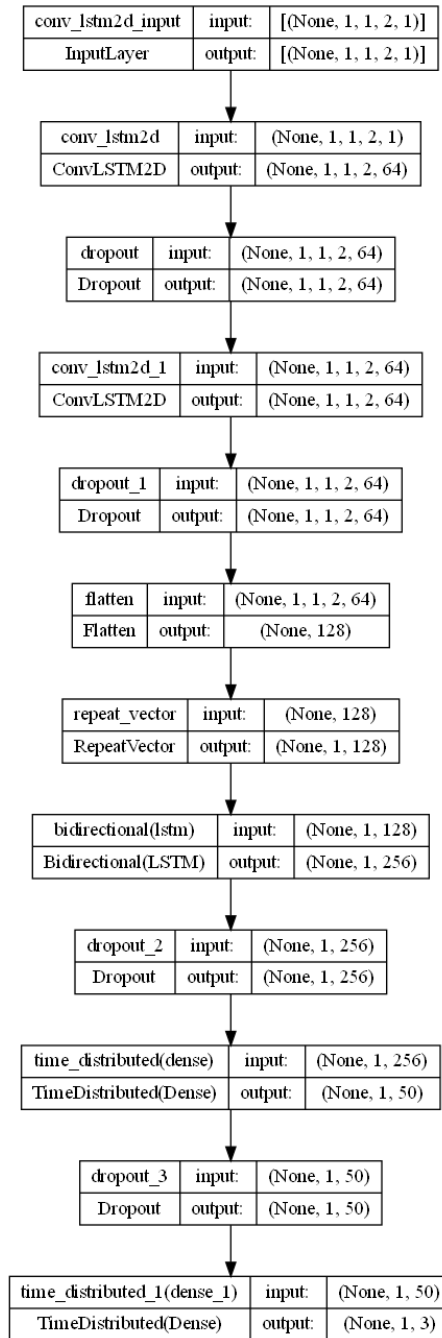


FIGURE 3.5 – Architecture du modèle Bi-ConvLSTM

L'entraînement du modèle est une étape cruciale dans le développement d'un modèle de détection et prédiction performant. Après avoir initialisé les paramètres et optimisé les hyperparamètres, le modèle apprend à travers les données d'entraînement. L'ajustement des poids des neurones est nécessaire dans ce processus afin de réduire l'erreur entre les prédictions et les valeurs réelles, ce qui améliore la capacité prédictive du modèle. On utilise fréquemment des méthodes de régularisation, comme le dropout, afin de prévenir le surapprentissage et d'améliorer la généralisation. Après avoir terminé l'entraînement,

le modèle est testé à l'aide d'un ensemble de données indépendant afin d'évaluer sa performance sur des données inconnues. Les paramètres du modèle sont entraînés à l'aide de l'algorithme de rétro-propagation à travers le temps (BPTT), qui met à jour les poids de manière itérative pour diminuer l'erreur de prédiction. Grâce à ce processus itératif, il est possible d'obtenir un modèle parfaitement adapté, capable de fournir des résultats précis et fiables dans des situations concrètes.

### 3.4.5 Test et évaluation du modèle

Une fois nos résultats de détection et prédiction obtenus, il est essentiel d'évaluer les performances de notre système à partir des données de test. Étant donné que notre modèle appartient à la catégorie des problèmes de régression, nous allons utiliser des métriques spécifiques à ce type de problème pour évaluer son efficacité. Les métriques que nous utiliserons pour évaluer les performances de notre modèle comprennent :

#### 3.4.5.1 La racine de l'erreur quadratique moyenne (RMSE) :

Le Root Mean Square Error (RMSE), ou erreur quadratique moyenne, est une mesure de l'écart entre les valeurs prédites par un modèle et les valeurs réelles. Elle est souvent utilisée comme norme pour mesurer les résultats de prédiction des modèles d'apprentissage automatique[45]. Formellement, le RMSE est calculé en prenant la racine carrée de la moyenne des carrés des écarts entre les valeurs prédites ( $\hat{y}$ ) et les valeurs réelles ( $y$ ) pour chaque observation dans un ensemble de données. Une petite valeur pour ce critère signifie que le modèle estimé est proche de la valeur réelle. La formule du RMSE est la suivante :

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (3.5)$$

Où

- $n$  est le nombre total d'observations,
- $y_i$  est la valeur réelle de l'observation  $i$ ,
- $\hat{y}_i$  est la valeur prédite pour l'observation  $i$ .

### 3.4.5.2 Erreur absolue moyenne (MAE) :

Le MAE (Mean Absolute Error) est une métrique utilisée pour évaluer la précision des modèles de régression. Il mesure la moyenne des valeurs absolues des écarts entre les valeurs prédites par le modèle et les valeurs réelles.[45] Une petite valeur pour ce critère signifie que le modèle estimé est proche de la valeur réelle. Formellement, le MAE est défini comme suit :

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (3.6)$$

Où

- $n$  est le nombre total d'observations,
- $y_i$  est la valeur réelle de l'observation  $i$ ,
- $\hat{y}_i$  est la valeur prédite pour l'observation  $i$ .

### 3.4.5.3 Recall

Le recall, aussi connu sous le nom de rappel ou de taux de rappel, est une mesure classante qui permet d'évaluer la capacité d'un modèle à repérer tous les exemples positifs. La proportion d'exemples positifs réels correctement identifiés parmi tous les exemples positifs est évaluée.[46]Le recall varie de 0 à 1, où :0 signifie que le modèle ne parvient pas à identifier les exemples positifs réels. 1 signifie que le modèle identifie tous les exemples positifs réels sans faux négatifs.Un recall élevé indique que le modèle est capable de trouver la plupart des exemples positifs, ce qui est important dans les cas où la sensibilité à la détection des positifs est critique.[53]La formule du rappel est donnée par :

$$\text{Rappel} = \frac{TP}{TP + FN} \quad (3.7)$$

Où :

- TP représente le nombre de vrais positifs .
- FN représente le nombre de faux négatifs.

### 3.4.5.4 F-score

Le F-score, aussi connu sous le nom de F1-score, est une métrique qui combine le rappel (recall) et la précision (precision) d'un modèle de classification en un seul point.

Il offre une évaluation globale des performances du modèle en prenant en considération à la fois les fausses bonnes et mauvaises choses. Le F-score atteint sa meilleure valeur à 1 (précision parfaite et rappel parfait) et sa pire valeur à 0.[46] [53] Un F-score élevé indique à la fois une précision et un rappel élevés, ce qui signifie que le modèle est capable de classer correctement les exemples positifs tout en minimisant à la fois les faux positifs et les faux négatifs. La formule du F1-score est la suivante :

$$F1 = 2 \cdot \frac{\text{précision} \cdot \text{rappel}}{\text{précision} + \text{rappel}} \quad (3.8)$$

#### 3.4.5.5 Précision

La précision (accuracy) est définie comme le rapport entre le nombre de prédictions correctes et le nombre total de prédictions effectuées.[46] Elle mesure la proportion de prédictions positives correctes parmi toutes les prédictions positives effectuées par le modèle. La formule de la précision est la suivante :

$$\text{Précision} = \frac{\text{Vrai Positifs (VP)}}{\text{Vrai Positifs (VP)} + \text{Faux Positifs (FP)}} \quad (3.9)$$

## 3.5 Conclusion

Ce chapitre présente de manière approfondie la conception de détection et prédiction hyper-paramétrée, incluant l'identification du jeu de données, sa découpe et son analyse, le prétraitement et la sélection des caractéristiques, ainsi que la définition de la méthode de recherche par grille et l'apprentissage du modèle jusqu'à l'obtention des résultats de détection et prédiction. L'étude expérimentale de notre travail sera traitée dans le chapitre suivant, englobant la mise en place de notre système, les outils matériels et logiciels employés, ainsi que le langage de développement du modèle. Pour terminer ce chapitre, nous exposerons les résultats obtenus et les comparerons afin de déterminer la valeur idéale des hyper-paramètres.

# Chapitre 4

## Simulation et Evaluation de Performances

### 4.1 Introduction

Ce chapitre présente une description de notre environnement de travail, du langage de programmation et des outils que nous avons utilisés pour mener à bien notre projet. Par la suite, nous exposerons les expérimentations conduites sur la méthode proposée ainsi que la discussion des résultats obtenus.

### 4.2 Environnement de travail

Les principales caractéristiques de la machines utilisée pour implémenter et tester notre approche sont :

- Type de la machine : DELL
- Processeur : Intel Core i5-2520M CPU @ 2.50GHz
- RAM : 6 Go
- Système d'exploitation : Windows 10 (64 bits)

### 4.3 Outils et bibliothèques utilisés

— **Anaconda :**

est une plateforme de distribution gratuite et open-source des langages de programmation Python et R, spécialisée dans le calcul scientifique, y compris la science des données, les applications d'apprentissage automatique, le traitement de données à grande échelle et l'analyse prédictive. Créée par Peter Wang et Travis Oliphant en

2012, cette distribution comprend l'interpréteur Python et plusieurs packages dans le domaine de l'apprentissage automatique et de la science des données[54].

— **Jupyter :**

Jupyter est une application web open source permettant de créer et de partager des documents interactifs. Ces documents peuvent contenir du code en direct, des équations, des visualisations et des textes explicatifs. Jupyter prend en charge plusieurs langages de programmation, notamment Python[54].

— **Python :**

Python est un puissant langage de programmation interprété et orienté objet. Le fait qu'il soit facile à apprendre et à utiliser le rend très apprécié pour une grande diversité de tâches de programmation, telles que la science des données, la création de scripts et la programmation Web[55].

— **Numpy :**

Numpy est une bibliothèque Python pour le calcul numérique. Elle offre des structures de données permettant de représenter des tableaux multidimensionnels et des matrices, ainsi que des fonctions mathématiques pour manipuler ces tableaux[56]. On utilise fréquemment Numpy dans les domaines de la science des données, de l'apprentissage automatique et du traitement d'images .

— **Pandas :**

Pandas est une bibliothèque Python qui permet de manipuler des données. Elle offre des structures de données permettant de représenter des tableaux de données, connus sous le nom de DataFrames, ainsi que des fonctionnalités pour les manipuler, les nettoyer et les analyser[57]. Pandas est employé dans le domaine de la science des données, de la finance et d'autres secteurs .

— **Matplotlib :**

Matplotlib est une bibliothèque Python conçue pour créer et visualiser des données sous forme de graphiques. Elle peut être intégrée avec les bibliothèques de calcul scientifique NumPy et SciPy, et elle est couramment utilisée dans divers environnements tels que les serveurs d'application web, les shells interactifs et les scripts Python[58].

---

— **Keras :**

Keras est une bibliothèque Python open source, à la fois puissante et conviviale, destinée au développement et à l'évaluation de modèles d'apprentissage profond[59]. Elle intègre les bibliothèques de calcul numérique performantes que sont Theano et TensorFlow, permettant ainsi de définir et d'entraîner des modèles de réseaux neuronaux en seulement quelques lignes de code .

— **Tensorflow :**

TensorFlow est une bibliothèque open source d'apprentissage automatique développée par Google, qui représente la deuxième génération du système de Google Brain. Elle permet de créer et d'exécuter des applications d'apprentissage automatique et profond[60]. TensorFlow sert de boîte à outils pour résoudre des problèmes mathématiques extrêmement complexes et pour effectuer des calculs numériques de haute performance avec aisance. Grâce à son architecture flexible, il est facile de déployer les calculs sur une variété de plateformes, notamment les processeurs CPU, les processeurs graphiques GPU et les unités de traitement tensoriel TPU .

— **Scikit-learn (Sklearn) :**

Scikit-learn est une bibliothèque logicielle open-source très prisée en machine learning pour le langage de programmation Python. Elle propose une large gamme d'algorithmes d'apprentissage automatique, qu'ils soient supervisés ou non supervisés, ainsi que des outils pour la préparation et la transformation des données, la sélection de modèles, l'évaluation des performances, et bien plus encore[61]. Scikit-learn est couramment utilisé pour le développement de modèles prédictifs, la réalisation de tâches de classification, de régression, de regroupement (clustering) et de réduction de dimensionnalité. Cette bibliothèque est appréciée pour sa convivialité, sa documentation complète et sa communauté active .

## 4.4 Simulation et résultats :

Cette partie est consacrée à l'analyse des performances de notre proposition. Pour ce faire, nous effectuerons des simulations en utilisant le langage de programmation Python. Nos données seront extraites d'un dataset open source provenant du suivi du trafic de Chicago, et seront prétraitées et analysées à l'aide de bibliothèques Python telles que Pandas, NumPy et Scikit-learn. Ces données incluent des informations sur la vitesse des véhicules, la longueur des segments, le nombre de véhicule et la distance moyenne entre les véhicules, et seront utilisées pour entraîner et évaluer notre modèle (Bi-ConvLSTM)

d'apprentissage profond.

#### 4.4.1 Hyperparamètres :

Les hyperparamètres sont essentiels pour le développement et la performance des modèles d'apprentissage profond. Contrairement aux paramètres du modèle, appris pendant l'entraînement, les hyperparamètres sont définis avant le début de l'entraînement et leur optimisation est cruciale pour la généralisation du modèle. Dans notre modèle (Bi-ConvLSTM), les hyperparamètres ont été ajustés manuellement à travers diverses expérimentations. Initialement, nous avons utilisé deux couches cachées de ConvLSTM2D avec différentes configurations de taille de lot, nombre d'époques, nombre de filtres, et taille des noyaux de convolution, mais la perte n'a pas diminué de manière significative, même avec une troisième couche ConvLSTM2D.

Ensuite, nous avons introduit un module LSTM bidirectionnel, en utilisant 256 unités, ce qui a produit des résultats satisfaisants avec une taille de lot de 32. Cependant, l'ajout d'une deuxième couche de Bi-LSTM n'a pas amélioré les résultats. En ajustant le taux d'apprentissage à  $1e-4$ , nous avons observé une diminution significative du RMSE. Finalement, l'utilisation de 256 unités dans une couche Bi-LSTM combinée avec un taux d'apprentissage de  $1e-4$  a donné les meilleurs résultats en termes de RMSE pour notre modèle, les hyperparamètres sont illustré dans le tableau 4.1 .

Hyperparamètre	Valeurs utilisées
Taille du lot	32
Nombre d'époques	100
Nombre de filtres	64
Taille des noyaux	(1, 1)
Taux d'apprentissage	$1e-5$ , $1e-4$
Unités Bi-LSTM	256
Taux de dropout	0.3, 0.4

TABLE 4.1 – Hyperparamètres du modèle

#### 4.4.2 Résultats obtenus :

##### 4.4.2.1 Taux d'erreur(Loss)

Dans ce qui suit, nous présentons les résultats de la simulation de notre modèle (Bi-ConvLSTM) d'apprentissage automatique.(figure 4.1).

La figure 4.1 présente le graphique qui montre l'évolution de la perte (loss) pour l'entraînement et la validation d'un modèle d'apprentissage automatique au cours de



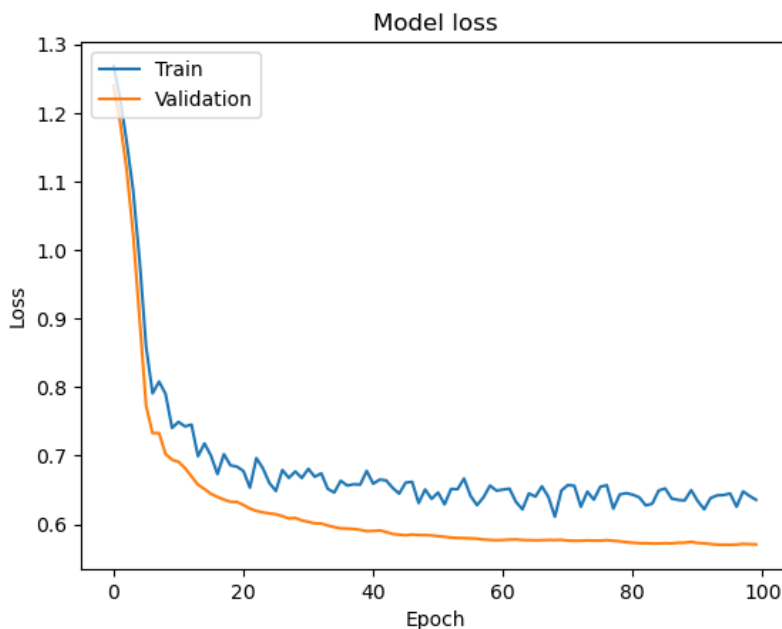


FIGURE 4.1 – la variation de l’erreur (Loss) du notre modèle Bi-ConvLSTM.

100 epochs, permettant ainsi d’évaluer la performance et la généralisation du modèle.

L’axe horizontal (X) représente le nombre d’epochs, qui sont les cycles complets pendant lesquels le modèle est entraîné sur l’ensemble des données. L’axe vertical (Y) représente la perte (loss), une métrique qui quantifie l’erreur du modèle. Une perte plus faible indique que le modèle prédit les données avec plus de précision.

La courbe de perte d’entraînement (ligne bleue) diminue rapidement au début, indiquant une phase d’apprentissage efficace où le modèle réduit significativement son erreur de prédiction sur les données d’entraînement. Cette diminution rapide est suivie par une stabilisation autour de valeurs constantes avec de légères oscillations, ce qui suggère que le modèle atteint une convergence après environ 20 epochs. Ces oscillations sont normales et reflètent les ajustements continus faits par l’algorithme d’optimisation.

La courbe de perte de validation (ligne orange) suit une tendance similaire avec une diminution rapide initiale, mais elle continue à baisser légèrement après la stabilisation de la perte d’entraînement et reste constamment inférieure à cette dernière. Ce comportement peut sembler contre-intuitif mais est souvent le signe d’un modèle bien régularisé, capable de généraliser efficacement aux données non vues grâce à des techniques comme le dropout, qui empêchent le surapprentissage. La perte de validation inférieure à la perte d’entraînement indique que le modèle apprend les caractéristiques pertinentes des données d’entraînement sans mémoriser le bruit ou les spécificités des données.

Les courbes montrent que le modèle apprend efficacement sans surapprentissage, ce qui implique qu'il peut détecter des cas avec une bonne précision et généralisation. La diminution continue et la stabilisation des pertes, associées à une meilleure performance sur les données de validation, confirment que le modèle est bien entraîné et apte à détecter les cas sur de nouvelles données de manière robuste et fiable.

En résumé, la diminution continue et la stabilisation des pertes, combinées à une meilleure performance sur les données de validation, montrent que le modèle est bien entraîné, régularisé de manière adéquate et possède une bonne capacité de généralisation. Cela est essentiel pour des performances robustes et fiables sur de nouvelles données, confirmant que le modèle n'est ni surentraîné ni sous-entraîné, mais bien équilibré pour prédire efficacement sur des ensembles de données inconnus.

#### 4.4.2.2 Analyse des résultats complémentaires

Afin d'évaluer de manière plus approfondie les performances de notre modèle (Bi-ConvLSTM), nous avons utilisé plusieurs métriques d'évaluation. Le tableau ci-dessous (Figure 4.2) présente les résultats obtenus :

Métrique	Valeur
F-Score	0.917
Rappel	0.944
Precision	0.9475
RMSE	0.2357
MAE	0.0556

TABLE 4.2 – Résultats des métriques d'évaluation

- **Précision** : La précision est définie comme le ratio des vraies prédictions positives sur le total des prédictions positives. Avec une précision de 0.9475, notre modèle a une performance très élevée. Cela signifie que près de 95% des instances que le modèle a classées comme positives sont effectivement positives.
- **Rappel** : Le rappel, ou sensibilité, mesure la capacité du modèle à identifier tous les échantillons positifs. Un rappel de 0.9444 montre que le modèle a identifié environ 94% des échantillons positifs réels. Cela indique que le modèle est efficace pour détecter les cas positifs et minimise les faux négatifs. Le rappel est crucial dans les situations où manquer un cas positif a des conséquences importantes.

- **F-score** : Un F-score de 0.9175, qui est la moyenne harmonique de la précision et du rappel, montre que le modèle maintient un bon équilibre entre ces deux métriques, ce qui est important pour obtenir une performance robuste en pratique.
- **MAE** : Une MAE de 0.0556 indique que les prédictions du modèle sont, très proches des valeurs réelles. Une MAE faible signifie que les erreurs du modèle sont faibles en termes de magnitude, ce qui est crucial pour des applications nécessitant une haute précision des prédictions individuelles.
- **RMSE** : Une RMSE de 0.2357, relativement faible, montre que les erreurs de prédiction du modèle sont également faibles en termes de variance.

Ces résultats de l'évaluation de notre modèle prédictif (Bi-ConvLSTM) sont très encourageants. La précision élevée, le rappel satisfaisant, ainsi que les faibles valeurs de MAE et de RMSE démontrent l'efficacité et la précision de notre modèle dans la prédiction des données. De plus, une exactitude de test élevée confirme sa capacité à généraliser sur de nouvelles données, soulignent la performance globale de notre modèle, ce qui en fait un outil fiable pour les tâches de prédiction.

## 4.5 Étude Comparative

Dans cette section, notre objectif est de démontrer l'efficacité de notre modèle amélioré (Bi-ConvLSTM) pour la détection et la prédiction de la congestion dans les systèmes de transport intelligent. Après des recherches approfondies, nous avons décidé de comparer notre modèle Bi-ConvLSTM avec deux approches couramment utilisées : un modèle LSTM simple et un modèle ConvLSTM2D-BiLSTM [44]. Pour effectuer cette comparaison, nous avons implémenté les deux méthodes citées précédemment en utilisant le même dataset (Paramètres de simulation). Cette comparaison a été effectuée selon le taux d'erreur et d'autres métriques de performance (RMSE, MAE, Précision, Rappel, F1-Score). Les résultats de ces comparaisons sont présentés ci-dessous.

### 4.5.1 Comparaison selon le taux d'erreur (Loss)

Après avoir implémenté les deux modèles LSTM et ConvLSTM2D-BiLSTM [44], nous avons évalué leurs taux d'erreur. Les résultats détaillés des modèles illustrés précédemment sont présentés dans les graphiques suivants (figure 4.2) :

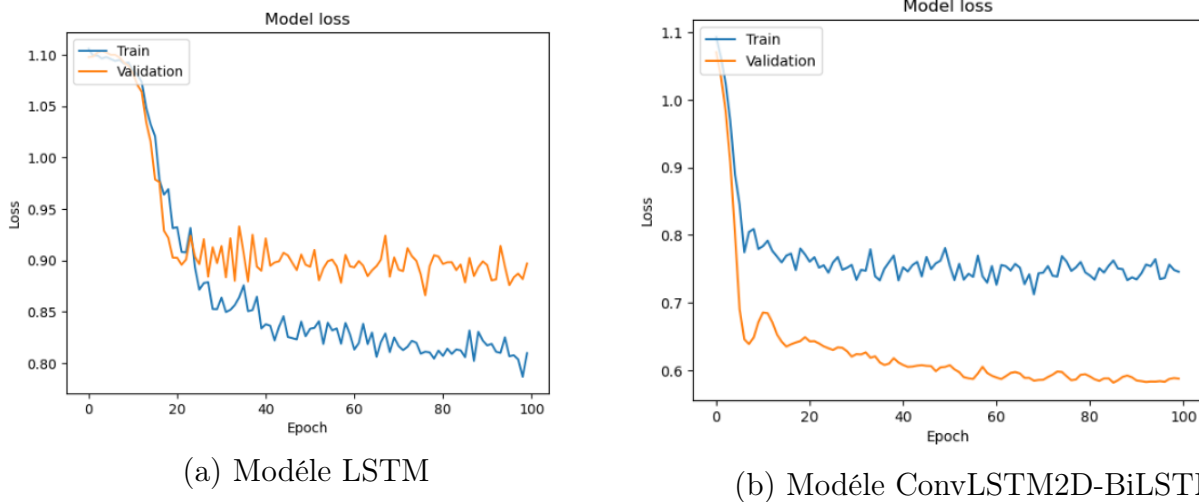


FIGURE 4.2 – La variation de l’erreur (Loss) des deux modèles.

Le premier graphique ( a ) couvre 100 époques, avec une perte initiale d’environ 1.10 et une perte finale d’entraînement d’environ 0.85. La perte de validation suit de près la perte d’entraînement, bien qu’elle présente quelques fluctuations, suggérant un léger besoin de régularisation ou d’ajustement des hyperparamètres pour améliorer la stabilité. Le modèle ne montre pas de signes significatifs de surapprentissage, car la perte de validation ne diverge pas de manière significative par rapport à la perte d’entraînement.

Le graphique ( b ) couvre 100 époques, avec une perte initiale également d’environ 1.10 et une perte finale d’entraînement d’environ 0.75. La perte de validation diminue plus rapidement que la perte d’entraînement et reste constamment inférieure, indiquant une bonne généralisation du modèle. Cette divergence positive suggère que le modèle apprend efficacement avec moins de risques de surapprentissage. Cependant, la courte durée de l’entraînement limite la capacité à évaluer la stabilité à long terme du modèle.

Notre graphique (figure 4.1) couvre 100 époques, avec une perte initiale plus élevée d’environ 1.25 et une perte finale d’entraînement d’environ 0.65. La perte de validation suit une tendance similaire mais reste constamment plus basse et lisse, indiquant une excellente généralisation et une absence de surapprentissage. La stabilité de la perte tout au long de l’entraînement, avec des variations minimales, suggère une bonne convergence et une capacité d’apprentissage robuste du modèle.

En conclusion, bien que tous les modèles montrent une réduction significative des pertes, le graphique (la figure 4.1) représente le meilleur modèle en termes de réduction globale des pertes, de stabilité et de généralisation. Il atteint les pertes les plus basses

et maintient un schéma constant sur une période d'entraînement plus longue sans signes de surapprentissage, indiquant une bonne performance et une excellente capacité de généralisation.

### 4.5.2 Comparaison selon les performances des modèles

Nous avons évalué le taux de (RMSE, MAE, Précision, Rappel, F-score) pour les deux modèles précédemment cités ainsi que notre modèle proposé (BiConvLSTM). Les résultats obtenus sont représentés sous forme de tableau (Table 4.3).

Modèle	RMSE	MAE	Précision	Rappel	F-score
LSTM	1.0	0.7	0.6	0.5	0.4
ConvLSTM2D-BiLSTM	0.8	0.5	0.7	0.6	0.6
BiConvLSTM	0.1	0.1	0.9	0.9	0.8

TABLE 4.3 – Comparaison des Métriques pour différents modèles

Afin de faciliter l'interprétation de ces résultats, nous avons également représenté le tableau sous forme de diagramme illustré dans (Figure 4.3).

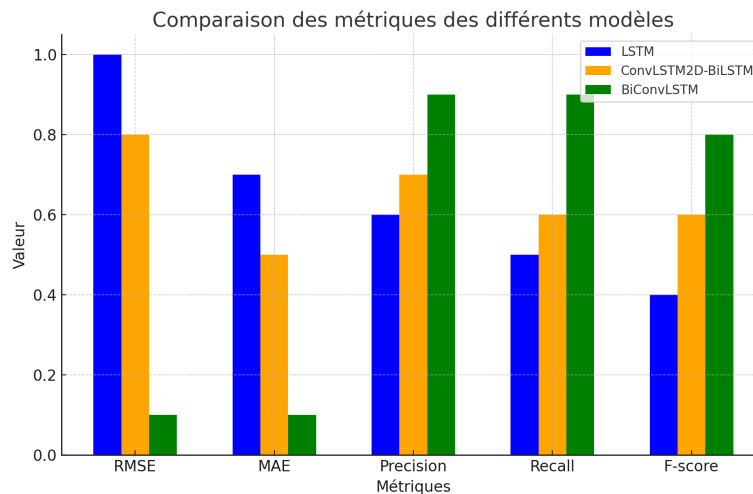


FIGURE 4.3 – Diagramme combiné des métriques des différents modèles

Dans cette analyse comparative des performances des trois modèles (LSTM, ConvLSTM2D-BiLSTM, BiConvLSTM) pour la détection et prédiction de la congestion du trafic, nous constatons que le modèle BiConvLSTM se distingue comme le plus performant parmi les trois. Ses faibles erreurs (RMSE de 0.1 et MAE de 0.1) montrent qu'il parvient à détecter et à prédire les valeurs avec une grande précision et minimise les écarts par rapport aux valeurs réelles. De plus, ses scores élevés en termes de précision (0.9), de rappel (0.9) et de F-score (0.8) indiquent une excellente capacité à identifier correctement les instances positives et à maintenir un équilibre optimal entre la précision et le rappel. Ces performances globales démontrent que le modèle BiConvLSTM est très efficace et fiable pour la tâche de détection et de prédiction de la congestion du trafic.

Le deuxième meilleur modèle, le ConvLSTM2D-BiLSTM, bien qu'inférieur au BiConvLSTM, présente tout de même des performances respectables. Son RMSE de 0.8 et son MAE de 0.5 indiquent une précision acceptable dans ses prédictions, même si ces valeurs sont plus élevées que celles du BiConvLSTM. En termes de précision (0.7), de rappel (0.6) et de F-score (0.6), le ConvLSTM2D-BiLSTM offre des résultats acceptables, suggérant qu'il peut encore être utilisé efficacement dans des contextes où une très haute précision n'est pas absolument nécessaire. Ses résultats montrent qu'il est capable de généraliser correctement les données, mais avec une marge d'erreur plus grande que le BiConvLSTM.

Enfin, le modèle LSTM se classe troisième, avec des performances moins satisfaisantes sur l'ensemble des métriques. Son RMSE de 1.0 et son MAE de 0.7 indiquent des erreurs plus élevées dans ses prédictions, ce qui peut affecter la fiabilité des résultats. En termes de précision (0.6), de rappel (0.5) et de F-score (0.4), le modèle LSTM montre des performances inférieures à celles des deux autres modèles. Ces scores inférieurs mettent en évidence les limites du LSTM simple pour la tâche spécifique de prédiction de la congestion du trafic, montrant qu'il est moins efficace pour capturer les dynamiques complexes nécessaires pour des prédictions précises.

En synthèse, l'évaluation des performances des trois modèles démontre que le BiConvLSTM est le modèle le plus performant et le plus fiable pour la prédiction de la congestion du trafic. Le ConvLSTM2D-BiLSTM se positionne comme une alternative fiable avec des performances décentes, tandis que le modèle LSTM, bien qu'utilisable, présente des performances significativement inférieures. Ces résultats suggèrent que, pour obtenir des prédictions précises et fiables, le modèle BiConvLSTM est à privilégier.

## 4.6 Conclusion

Ce chapitre offre une vue d'ensemble complète de l'implémentation et de l'évaluation des performances de notre modèle en mettant en évidence les choix de conception, les ajustements d'hyperparamètres et les résultats obtenus à partir des simulations. Nous avons également présenté une analyse comparative des performances de notre modèle par rapport à des modèles existants. Les résultats montrent que notre modèle amélioré offre de meilleures performances en termes de précision, de rappel, de F-score, de RMSE et de MAE. Cela confirme l'efficacité de notre approche pour la détection et la prédiction de la congestion dans les systèmes de transport intelligent.

# Conclusion générale et perspectives

Les systèmes de transport intelligents (STI) sont devenus essentiels pour moderniser la gestion du trafic urbain et améliorer la mobilité. Cependant, la congestion routière reste un problème persistant, surtout dans les zones urbaines à forte densité de population. Les embouteillages entraînent des pertes de temps, une augmentation de la pollution et des risques accrus d'accidents.

Pour aborder ces défis, nous avons mené une étude bibliographique approfondie sur les différentes approches de gestion du trafic existantes. Cette recherche nous a permis de comprendre les avantages et les limitations des méthodes actuelles. Sur cette base, nous avons développé une nouvelle approche utilisant des réseaux de neurones convolutifs bidirectionnels et des modèles de mémoire à long terme (Bi-ConvLSTM). Cette méthode combine les techniques de détection et de prédiction de la congestion, offrant une solution plus robuste et efficace.

Nous avons également effectué une étude comparative des différentes méthodes de gestion de la congestion. Les résultats montrent que notre modèle Bi-ConvLSTM surpasse les approches traditionnelles en termes de précision et de rapidité de détection et de prédiction des conditions de trafic. Les simulations et les tests effectués démontrent que notre modèle réduit significativement les taux d'erreur et améliore la fluidité du trafic.

Nous présentons des perspectives pour nos futures recherches. Nous envisageons d'améliorer notre solution en intégrant des données supplémentaires, telles que les actions du conducteur et les conditions de la route. De plus, nous prévoyons d'inclure des scénarios réels de conduite pour tester davantage notre modèle et perfectionner ses capacités de prédiction et de détection. Enfin, nous ambitionnons de développer des systèmes capables de fournir des recommandations en temps réel aux conducteurs afin d'améliorer la sécurité et l'efficacité des transports intelligents.

L'importance de ce travail réside dans son potentiel à transformer la gestion du trafic urbain. En réduisant la congestion, notre approche contribue non seulement à amélio-



rer la qualité de vie des citoyens mais aussi à diminuer l'impact environnemental des transports. Ainsi, ce projet participe à la création de villes plus intelligentes, fluides et durables, répondant aux besoins croissants de mobilité tout en préservant l'environnement.

# Bibliographie

- [1] Pramila P Shinde and Seema Shah. A review of machine learning and deep learning applications. In *2018 Fourth international conference on computing communication control and automation (ICCCUBEA)*, pages 1–6. IEEE, 2018.
- [2] Hojjat Salehinejad, Sharan Sankar, Joseph Barfett, Errol Colak, and Shahrokh Valaee. Recent advances in recurrent neural networks. *arXiv preprint arXiv :1801.01078*, 2017.
- [3] Tanya Garg and Gurjinder Kaur. A systematic review on intelligent transport systems. *Journal of Computational and Cognitive Engineering*, 2(3) :175–188, 2023.
- [4] Javier J. Sanchez-Medina, Eleni I. Vlahogianni, and Javier Del Ser. From data to actions in intelligent transportation systems : A prescription of functional requirements for model actionability. *Sensors*, 21(4) :1121, 2021.
- [5] Antoine B. Bagula, Hloniphani C. Maluleke, and Isaac A. Odun-Ayo. Transport inequalities and the adoption of intelligent transportation systems in africa : A research landscape. *Sustainability*, 13(22) :12891, 2021.
- [6] Li Zhang, Hao Wang, and Jie Zhao. Evaluation of intelligent transportation systems : Methodologies and frameworks. *Journal of Traffic and Transportation Engineering*, 8(3) :355–369, 2021.
- [7] Tao Zhang, Hyo-Sang Shin, Xiang Xu, and Gaoge Hu. Recent advances in intelligent transportation systems. *Electronics*, 10(1) :101, 2021.
- [8] Journal Général de l’Europe. Systèmes de transport intelligents - journal général de l’europe. *Journal Général de l’Europe*, 2024.
- [9] Techno-Science.net. Système de transport intelligent : définition et explications. *Techno-Science.net*, 2024.
- [10] Grimard. Systèmes de transport intelligents (sti) - grimard. *Grimard*, 2024.
- [11] IFP Énergies Nouvelles. Système de transport intelligent et mobilité 3.0 : Définition, enjeux et acteurs. Récupéré sur <https://www.ifpennergiesnouvelles.fr/article/systeme-transport-intelligent-et-mobilite30-definition-enjeux-et-acteurs>, s.d.

- 
- [12] Road Network Operations. Road network operations. <http://road-network-operations.piarc.org/index.php?option=comcontent&task=view&id=39&Itemid=71&lang=fr>. Accessed : 2024-06-09.
- [13] F. Doetzer, F. Kohlmayer, T. Kosch, and M. Strassberger. Secure communication for intersection assistance. In *Proc. of the 2nd Int. Workshop on Intelligent Transportation*, Hamburg, Germany, March 2005.
- [14] A. Cornuéjols, L. Miclet, and Y. Kodratoff. *Apprentissage Artificiel, Concepts et algorithmes*. Dunod, 2002.
- [15] Fadhila Guenez. *Modèle d'application du problème d'ordonnancement des tâches dans les systèmes distribués résolvant les Problèmes de planification de la circulation routière dans les villes intelligentes*. PhD thesis, Université Larbi Tébessi-Tébessa, 2022.
- [16] Ranwa Al-Mallah. *Road Traffic Congestion Analysis Via Connected Vehicles*. PhD thesis, École Polytechnique de Montréal, 2018.
- [17] Mariette Awad and Rahul Khanna. *Efficient learning machines : theories, concepts, and applications for engineers and system designers*. Springer nature, 2015.
- [18] Batta Mahesh. Machine learning algorithms-a review. *International Journal of Science and Research (IJSR).[Internet]*, 9(1) :381–386, 2020.
- [19] Jason Brownlee. 14 different types of learning in machine learning, 2019.
- [20] Yisheng Lv, Yanjie Duan, Wenwen Kang, Zhengxi Li, and Fei-Yue Wang. Traffic flow prediction with big data : A deep learning approach. *Ieee transactions on intelligent transportation systems*, 16(2) :865–873, 2014.
- [21] Basharat Hussain, Muhammad Khalil Afzal, Shafiq Ahmad, and Almetwally M Mostafa. Intelligent traffic flow prediction using optimized gru model. *IEEE Access*, 9 :100736–100746, 2021.
- [22] Shuai Gao, Yuefei Huang, Shuo Zhang, Jingcheng Han, Guangqian Wang, Meixin Zhang, and Qingsheng Lin. Short-term runoff prediction with gru and lstm networks without requiring time step optimization during sample generation. *Journal of Hydrology*, 589 :125188, 2020.
- [23] Christopher Olah. Understanding lstm, 2015.
- [24] Balachandran Vijayalakshmi, Kadarkarayandi Ramar, NZ Jhanjhi, Sahil Verma, Madasamy Kaliappan, Kandasamy Vijayalakshmi, Shanmuganathan Vimal, Kavita, and Uttam Ghosh. An attention-based deep learning model for traffic flow prediction using spatiotemporal features towards sustainable smart city. *International Journal of Communication Systems*, 34(3) :e4609, 2021.
- [25] Saurabh Rathor. Simple rnn vs gru vs lstm : Difference lies in more flexible control, 2018. Web page.

- 
- [26] Guizhu Shen, Qingping Tan, Haoyu Zhang, Ping Zeng, and Jianjun Xu. Deep learning with gated recurrent unit networks for financial sequence predictions. *Procedia computer science*, 131 :895–903, 2018.
- [27] Keping Liang, Long Lin, Mamoun Alazab, Liang Tan, and Bo Gu. A deep learning approach to traffic lights control for autonomous vehicles. *IEEE Transactions on Intelligent Transportation Systems*, pages 381–390, 2020.
- [28] Cheng. Real-time vehicle detection and tracking algorithm based on yolov3. In *2021 IEEE International Conference on Mechatronics, Automation and Intelligent Control (ICMAIC)*, pages 1–6. IEEE, 2021.
- [29] Subramanian Du and Bhargavi. Driving behavior analysis based on machine learning : A survey. In *IEEE Access*, 7, 150193-150215. IEEE.
- [30] Cosimo Pereira, Paola Panuccio, and da Silva M Rindone, Corrado. Urban mobility planning : A review of methodologies and tools. *transport reviews*. pages 592–611.
- [31] Sara Li, Moridpour, Alireza Toran Pour, and Tayebbeh Saghapour. Big data analytics for traffic and transportation engineering : Applications, challenges, and opportunities. pages 22122–22138. IEEE, 2020.
- [32] Kumar Kaur. A review on security and privacy issues in intelligent transportation systems. pages 2143–2148. IEEE, 2021.
- [33] Jianghui Zhang, Jinping Liu, and Fang-Mei Tseng. The development of smart transportation in china : Policies, technologies, and challenges. *technological forecasting and social change*. pages 142, 56–69. IEEE, 2019.
- [34] Amit Kumar Wang and Niladhuri Sreenath. Intelligent transportation systems : Past, present, and future. pages 2966–2986. IEEE, 2020.
- [35] Md Ashifuddin Mondal and Zeenat Rehena. Intelligent traffic congestion classification system using artificial neural network. In *Companion Proceedings of The 2019 World Wide Web Conference*, pages 110–116, 2019.
- [36] Sanaz Shaker Sepasgozar and Samuel Pierre. Network traffic prediction model considering road traffic parameters using artificial intelligence methods in vanet. *IEEE Access*, 10 :8227–8242, 2022.
- [37] LMI Leo Joseph, Pankaj Goel, Ashish Jain, K Rajyalakshmi, Kamal Gulati, and Prithvipal Singh. A novel hybrid deep learning algorithm for smart city traffic congestion predictions. pages 561–565, 2021.
- [38] Xin Wang, Jing Liu, Tie Qiu, Chaoxu Mu, Chen Chen, and Pan Zhou. A real-time collision prediction mechanism with deep learning for intelligent transportation system. *IEEE transactions on vehicular technology*, 69(9) :9497–9508, 2020.

- 
- [39] Meng Chen, Xiaohui Yu, and Yang Liu. Pcn : Deep convolutional networks for short-term traffic congestion prediction. *IEEE Transactions on Intelligent Transportation Systems*, 19(11) :3550–3559, 2018.
- [40] Fei-Rong Huang, Cong-Xiang Wang, and Chih-Min Chao. Traffic congestion level prediction based on recurrent neural networks. In *2020 International conference on artificial intelligence in information and communication (ICAIIIC)*, pages 248–252. IEEE, 2020.
- [41] Dong-Fan Xie, Zhe-Zhe Fang, Bin Jia, and Zhengbing He. A data-driven lane-changing model based on deep learning. *Transportation research part C : emerging technologies*, 2019.
- [42] Md Mohiuddin Chowdhury, Mahmudul Hasan, Saimoom Safait, Dipankar Chaki, and Jia Uddin. A traffic congestion forecasting model using cmtf and machine learning. pages 357–362, 2018.
- [43] Tariku Sinshaw Tamir, Gang Xiong, Zhishuai Li, Hao Tao, Zhen Shen, Bin Hu, and Heruye Mulugeta Menkir. Traffic congestion prediction using decision tree, logistic regression and neural networks. *Ifac-PapersOnline*, 2020.
- [44] Tao Li, Anning Ni, Chunqin Zhang, Guangnian Xiao, and Linjie Gao. Short-term traffic congestion prediction with conv-bilstm considering spatio-temporal features. *IET Intelligent Transport Systems*, 14(14) :1978–1986, 2020.
- [45] Initiez-vous au machine learning, 2024.
- [46] Abdelghani BAKKI, Abdelhafid MABROUKI, Abd El Ouahabe Ouahabe, et al. *L'apprentissage profond pour la classification des zones brûlées après un feu de forêts à partir des images satellitaires*. PhD thesis, UNIVERSITE AHMED DRAIA-ADRAR, 2023.
- [47] Tanzina Afrin and Nita Yodo. A survey of road traffic congestion measures towards a sustainable and resilient transportation system. *Sustainability*, 12(11) :4660, 2020.
- [48] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn : Machine learning in python. *the Journal of machine Learning research*, 12 :2825–2830, 2011.
- [49] Adolf Darlington May. *Traffic flow fundamentals*. 1990.
- [50] Markos Papageorgiou, Christina Diakaki, Vaya Dinopoulou, Apostolos Kotsialos, and Yibing Wang. Review of road traffic control strategies. *Proceedings of the IEEE*, 91(12) :2043–2067, 2003.
- [51] Martin Treiber, Arne Kesting, Martin Treiber, and Arne Kesting. Model-based traffic flow optimization. *Traffic Flow Dynamics : Data, Models and Simulation*, pages 403–422, 2013.

- 
- [52] Letty Aarts and Ingrid Van Schagen. Driving speed and the risk of road crashes : A review. *Accident Analysis & Prevention*, 38(2) :215–224, 2006.
- [53] Latifa Bekkar, Safa Goucem, and Seddik Encadre par Khemaissia. *Conception d'un modèle d'apprentissage profond pour la détection des maladies cancéreuses*. PhD thesis, 2023.
- [54] Project jupyter. 1/06/2023.
- [55] Python.org. what is python? executive summary. 20/05/2024.
- [56] Numpy. 20/05/2024.
- [57] pandas documentation. 20/05/2024.
- [58] matplotlib. 20/05/2024.
- [59] keras. 20/05/2024.
- [60] Tensorflow documentation. 20/05/2024.
- [61] scikit-learn : machine learning in python — scikit-learn 1.2.2 documentation. 20/05/2024.

---

## RÉSUMÉ

Dans un contexte où les avancées technologiques sont constantes, la gestion efficace et sécurisée des systèmes de transport devient primordiale. L'utilisation de systèmes de transport intelligents est désormais essentielle dans la vie quotidienne, ce qui rend la prédiction et la détection des anomalies dans ces systèmes cruciales. L'objectif principal de ce mémoire est de proposer une solution pour la prédiction et détection de l'état du trafic à l'aide de techniques d'intelligence artificielle. Pour atteindre cet objectif, nous avons développé une méthode de prédiction et détection basée sur des modèles de réseaux de neurones convolutifs (CNN) et des mémoires à long terme (LSTM). Un système expérimental a été conçu pour permettre la simulation de ces techniques. Les résultats obtenus démontrent la faisabilité et l'efficacité de notre proposition. Nos recherches futures incluent l'amélioration du système par l'intégration de données supplémentaires sur les actions des conducteurs et les conditions de la route, ainsi que le développement de systèmes capables de fournir des recommandations en temps réel.

**Mots clés :** Prédiction de l'état du trafic ; Systèmes de Transport Intelligent ; Réseaux de Neurones Récurrents ; Mémoires à Long Terme ; Intelligence Artificielle.

## ABSTRACT

In a context where technological advancements are constant, the effective and secure management of transportation systems becomes paramount. The use of intelligent transportation systems is now essential in daily life, making the prediction and detection of anomalies in these systems crucial. The primary objective of this thesis is to propose a solution for traffic state prediction using artificial intelligence techniques. To achieve this objective, we developed a prediction method based on convolutional neural networks (CNN) and long short-term memory (LSTM) models. An experimental system was designed to simulate these techniques. The results obtained demonstrate the feasibility and effectiveness of our proposal. Our future research includes improving the system by integrating additional data on driver actions and road conditions, as well as developing systems capable of providing real-time recommendations.

**Key words :** Traffic State Prediction ; Intelligent Transportation Systems ; Recurrent Neural Networks ; Long Short-Term Memory ; Artificial Intelligence.