

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 MASTER RECHERCHE

En

Informatique

Option

Intelligence Artificielle

Thème

Proposition d'un modèle de prédiction de la demande par
Machine Learning pour l'optimisation de la chaîne logistique

Présenté par : Mlle. AIT RADI Ouiza & Mlle. ALLOU Nadia

Soutenu le 04 Juillet 2024 devant le jury composé de :

Présidente	Dr BOUKERRAM Samira
Encadrante	Dr BOUKREDERA Djamila
Examineur	Dr ATMANI Mouloud

Béjaïa, Juillet 2024.

※ *Remerciements* ※

En premier lieu nous remercions Dieu le tout puissant pour toute la volonté et le courage qu'il nous a donné pour l'achèvement de ce travail.

Nous tenons à remercier très chaleureusement, **Dr BOULAHROUZ** pour la qualité de l'encadrement, dont elle nous a fait bénéficier, pour avoir mieux guidé et structurer ce travail en conjuguant habilement, disponibilité, conseils et critiques constructives. Ainsi que de nous avoir fait profiter de son expérience.

Toutes nos sincères gratitudee et notre profond respect à **Dr BOUKERRAM** qui nous a honoré en acceptant de présider notre soutenance.

Nous remercions vivement **Dr ATMANI** qui a accepté d'examiner et de valoriser notre travail.

Nous tenons à remercier l'ensemble des enseignants du département informatique ayant contribué à notre formation durant notre cycle d'étude.

Mlle. ALLOU Nadia & Mlle. AIT RADI Ouiza

※ *Dédicaces* ※

A mes chers parents, aucune dédicace ne saurait être assez éloquente pour vous exprimer ce que vous méritez pour tous les efforts et les sacrifices que vous n'avez jamais cessé de consentir pour mon instruction et mon bien-être. Je vous rends hommage par ce modeste travail en guise de ma reconnaissance éternelle et de mon infini amour.

A ma très chère petite nièce Imane « Ili Dunith », et mon cher neveu Ilyan, que dieu vous protège.

A mes grands-parents, mes sœurs Lili, Hanane et Lyna, mon frère Mami, mon oncle Fahim et à toute ma famille... Les mots ne suffisent guère pour exprimer l'attachement, l'amour et l'affection que je vous porte.

A ma binôme Nadia, je te souhaite un bon avenir.

A toutes mes amies : Asma, Nesrine, Melissa, Lydia, Zouina, Ghania, Kenza et ceux qui me sont chers, la liste est bien longue.

Mlle AIT RADI Ouiza

※ *Dédicaces* ※

À **mes très chers parents**, qui m'ont soutenue à chaque étape de ma vie et qui ont toujours cru en moi, je dédie ce travail.

À mon petit **frère** et **mes sœurs** bien-aimés. En particulier, à ma très chère sœur **Hanane** qui n'a jamais cessé de croire en moi et de m'encourager à être à la hauteur. Merci infiniment pour ton soutien indéfectible tout au long de mon parcours universitaire.

À chaque membre de **ma famille** qui m'a toujours encouragée, un immense merci pour leur amour et leur soutien.

À ma binôme **Ouiza**, merci énormément pour ta collaboration.

À des personnes spéciales :

- Mon cher oncle **BIR Boubekar**, qui ne cesse de me motiver et de me soutenir en toutes circonstances. Merci infiniment d'être comme un grand frère bienveillant et protecteur.
- Ma chère tante **BIR Samia**, toujours présente à mes côtés durant mon parcours universitaire, et me rappelant constamment de mes capacités. Merci énormément pour ton soutien constant.
- Mr **MIZI ALLAOUA Farid**, pour son accueil chez Cevital Agro-Industrie. Je le remercie infiniment pour ses encouragements et son encadrement qui m'ont permis d'apprendre sur la chaîne logistique, une partie essentielle de mon PFE.

À toutes **les bonnes personnes** et **amies sincères** que j'ai rencontrées pendant mes études, dont la compagnie et le soutien ont été des sources de joie et de motivation.

Mlle. ALLOU Nadia

Table des matières

Table des matières	iv
Table des figures	vi
Liste des tableaux	vii
Liste des abréviations	ix
Introduction générale	1
1 Généralités sur l'optimisation de la chaîne logistique	3
1.1 Introduction	3
1.2 Chaîne logistique	4
1.2.1 Définition et composantes de la chaîne logistique	4
1.2.2 Structure de la chaîne logistique	5
1.2.3 Les flux de la chaîne logistique	6
1.2.4 Les processus de la chaîne logistique	7
1.3 La gestion de la chaîne logistique (SCM : Supply Chain Management)	8
1.3.1 Définition de gestion de la chaîne logistique	8
1.3.2 Les tâches dédiées au service SCM	9

1.3.3	Les défis courants rencontrés dans la gestion de la chaîne logistique	10
1.4	L'optimisation de la chaîne logistique	11
1.4.1	Objectifs de l'optimisation de la chaîne logistique	11
1.4.2	Méthodes traditionnelles d'optimisation de la chaîne logistique et leurs limites . . .	12
1.4.3	Contribution de l'IA à la résolution des défis de la chaîne logistique	13
1.5	Conclusion	13
2	Etat de l'art sur l'IA et la prévision de la demande dans une chaîne logistique	14
2.1	Introduction	14
2.2	Prévision de la demande pour la gestion de la chaîne logistique	14
2.2.1	Rôle de la prévision de la demande	15
2.2.2	L'Intelligence Artificielle (IA) dans la prévision de la demande	16
2.3	Modèles de prédiction de la demande	16
2.3.1	Modèles simples	17
2.3.2	Modèles d'ensemble	22
2.3.3	Modèles hybrides	25
2.3.4	Différence entre les modèles hybrides et l'approche ensembliste	25
2.4	Les séries chronologiques	25
2.5	Métriques d'évaluation pour les modèles de prédiction	27
2.6	Travaux connexes	29
2.7	Tableau comparatif	33
2.8	Conclusion	34
3	Proposition d'un modèle de prédiction de la demande dans une chaîne logistique	35
3.1	Introduction	35
3.2	Dataset	35

3.3	Visualisation de données	36
3.4	Préparation de données	38
3.4.1	Nettoyage de données	38
3.4.2	Optimisation de l'analyse de la demande	38
3.4.3	Transformation des Données en Série Temporelle Supervisée	39
3.4.4	Fractionnement de Dataset	39
3.4.5	Normalisation de données	40
3.5	Modèle proposé	40
3.6	Implémentation	42
3.6.1	Environnement de développement	42
3.6.2	Langage de programmation :	43
3.6.3	Bibliothèques utilisées :	43
3.6.4	Hyperparamètres	44
3.6.5	Entraînement des modèles de base	46
3.6.6	Entraînement des modèles ensemblistes :	48
3.7	Evaluation des modèles	48
3.8	Résultats et discussion	50
3.8.1	Résultats des modèles individuels	50
3.8.2	Résultats des approches ensemblistes	50
3.8.3	Comparaison des résultats entre l'approche proposée et celle de l'article [41]	51
3.8.4	Valeurs réelles et valeurs prédites	53
3.8.5	Schéma récapitulatif de notre approche proposée	54
3.9	Conclusion	55
	Conclusion générale & perspectives	56

Bibliographie

58

Table des figures

1.1	Modèle de chaîne logistique [14]	4
1.2	Exemple de chaîne logistique [28]	4
1.3	Différentes structures de la chaîne logistique [44]	6
1.4	Les Flux de la chaîne logistique[6]	7
1.5	Supply Chain Management : Modèle illustratif[13].	9
2.1	Prévision de la demande et gestion des stocks [39]	15
2.2	Modèles de prédiction de la demande	17
2.3	Architecture d'un Réseau de Neurones Artificiels (ANN)	18
2.4	Structure de base d'un modèle CNN [12]	19
2.5	Structure de base d'un modèle LSTM [22]	21
2.6	Structure de base d'un modèle GRU. [1]	22
2.7	Illustration des méthodes d'ensemble en apprentissage automatique	22
2.8	Étapes du Boosting	23
2.9	Étapes du Bagging	23
2.10	Étapes du Stacking	24
2.11	Prévision de la demande en séries temporelles à l'aide d'un ensemble de deep learning [41]	29

3.1	Les premières lignes du jeu de données	36
3.2	La demande quotidienne	37
3.3	La demande quotidienne par magasin	37
3.4	La demande quotidienne par article	38
3.5	Répartition des Données en Ensembles d’Entraînement, de Validation et de Test.	40
3.6	Schéma du modèle ensembliste proposé avec CNN, GRU et XGBoost pour la prédiction de la demande.	42
3.7	Courbes de perte du Modèle GRU	49
3.8	Courbes de perte du Modèle CNN	49
3.9	Courbes de perte du Modèle MLP	49
3.10	Courbes de perte du Modèle LSTM	49
3.11	Erreur quadratique moyenne	51
3.12	Racine de l’erreur quadratique moyenne	51
3.13	Coefficient de Détermination R^2 (R-squared)	52
3.14	Résultats de prédiction sur les données de test	53
3.15	Méthodologie proposée de travail	54

Liste des tableaux

2.1	Résumé des travaux connexes	34
3.1	Performance des modèles de prédiction avec les métriques RMSE, MAE et R^2	50
3.2	Performance des Approches ensemblistes	50

Liste des abréviations

1D-CNN One-Dimensional Convolutional Neural Network

ANN Artificial Neural Network

ARIMA Autoregressive Integrated Moving Average

ARMA Autoregressive Moving Average

GRU Gated Recurrent Unit

IA Intelligence Artificielle

IDE Integrated Development Environment

LR Linear Regression

LSTM Long Short-Term Memory Network

MAE Mean Absolute Error

MAPE Mean Absolute Percentage Error

ML Machine Learning

MLP Multi-Layer Perceptron

MLR Multiple Linear Regression

MSE Mean Squared Error

RMSE Root Mean Squared Error

RNN Recurrent Neural Networks

SARIMA Seasonal Autoregressive Integrated Moving Average

SCM Supply Chain Management

Spyder Scientific Python Development Environment

XGBoost eXtreme Gradient Boosting

Introduction générale

Le défi le plus important adressé par les entreprises qui essaient d'être plus compétitives sur le marché mondial implique la gestion de la chaîne logistique. Cela inclut divers processus tels que l'approvisionnement, la distribution, la planification, la production et la gestion des stocks, parmi d'autres processus. La chaîne logistique est essentielle étant donné la volatilité accrue du marché et les besoins des consommateurs. Malgré leur efficacité dans certaines situations, les systèmes traditionnels de gestion de la chaîne logistique présentent des limites en termes de flexibilité, d'adaptabilité et de capacité à prendre des décisions en temps réel. Ces dernières années, la recherche s'est donc orientée vers le développement de méthodes innovantes basées sur le Machine Learning (ML) (Apprentissage Automatique) afin d'optimiser ces processus. L'émergence de l'intelligence artificielle (IA) et des techniques de ML ouvre de nouvelles perspectives pour transformer la gestion de la chaîne logistique. Les réseaux de neurones artificiels et les modèles de prévision basés sur les séries temporelles, par exemple, promettent d'améliorer significativement la précision des prévisions de la demande logistique, renforçant ainsi l'efficacité de la gestion des stocks.

Les modèles de ML offrent la capacité de révolutionner la gestion des stocks et la planification de la production. Grâce à leur capacité à traiter de vastes ensembles de données et à identifier des modèles complexes dans le comportement de la demande, ces outils permettent non seulement une prévision plus précise à court terme, mais aussi une anticipation proactive des tendances à long terme. Ces avancées promettent de transformer significativement la gestion des stocks, d'optimiser les niveaux de stocks, de réduire les coûts de transport et d'accroître la satisfaction client, consolidant ainsi la compétitivité des entreprises dans un marché globalisé de plus en plus complexe.

Dans le cadre de notre projet, nous développons une approche ensembliste pour la prévision de la demande au sein de la chaîne logistique en utilisant des modèles de ML avancés. Notre étude porte sur un ensemble de données couvrant cinq années de demandes quotidiennes pour 50 articles répartis dans 10 magasins distincts. Nous explorons trois architectures principales de réseaux neuronaux - GRU, 1D-CNN et XGBoost - pour notre approche proposée. Chacune de ces architectures contribue de manière distincte

à notre modèle prédictif.

Le modèle GRU que nous utilisons comprend deux couches GRU suivies d'une couche "Dense" en sortie. La première couche GRU, dotée de 64 unités et d'une activation ReLU, capture efficacement les dépendances temporelles. La seconde couche GRU, avec 32 unités et la même activation, résume les séquences en un vecteur de taille 32, optimisé par des couches de "Dropout" à 20% pour prévenir le surapprentissage. En parallèle, notre modèle 1D-CNN intègre une couche de convolution à une dimension (1D) avec 64 filtres et une fenêtre de taille 2, suivie d'une couche de MaxPooling1D pour réduire la dimensionnalité. La sortie est aplatie avant d'être traitée par deux couches "Dense" : la première, composée de 50 neurones avec activation ReLU, combine efficacement les caractéristiques extraites, tandis que la seconde génère la prédiction finale. Cette architecture permet de capturer les motifs locaux pertinents dans nos séries temporelles. En complément, nous introduisons un méta-modèle XGBoost qui fusionne les prédictions des modèles individuels GRU et 1D-CNN en tant que nouvelles caractéristiques d'entrée. Ce méta-modèle est entraîné sur un ensemble d'entraînement construit à partir des prédictions des modèles initiaux, consolidant ainsi les forces prédictives de chaque architecture.

Pour évaluer rigoureusement notre approche, nous utilisons plusieurs métriques clés telles que le RMSE (Root Mean Square Error), le MAE (Mean Absolute Error) et le coefficient de détermination R^2 (R-squared). Ces mesures nous permettent de quantifier la précision et la robustesse de notre modèle sur les données de test. Les résultats obtenus démontrent que notre approche surpasse significativement les benchmarks existants en termes de précision prédictive, avec un RMSE de 9,70 et un MAE de 7,19 comparé à 11,30 et 8,40 respectivement pour l'approche comparative.

Les résultats détaillés de nos évaluations montrent une nette amélioration ; notre approche a non seulement réduit les erreurs moyennes absolues et quadratiques, mais a également démontré une meilleure capacité à expliquer la variance des données observées, comme indiqué par un R^2 de 0,89 comparé à 0,84 pour l'approche comparative. Ces résultats consolidés confirment l'efficacité de notre méthodologie pour résoudre le problème spécifique de prédiction de la demande dans la chaîne logistique, promettant ainsi des avantages significatifs en termes de performance opérationnelle et de gestion des stocks.

En résumé, notre mémoire est structuré en trois chapitres principaux. Dans le premier chapitre, nous explorons les fondements de la chaîne logistique, sa gestion et son optimisation, ainsi que les principaux défis rencontrés dans ce domaine et l'impact potentiel de l'intelligence artificielle. Le chapitre 2 présente un état de l'art des méthodes et des divers modèles d'apprentissage automatique utilisés dans la littérature pour la prévision de la demande. Nous examinons les études de cas et les projets de recherche qui ont utilisé avec succès ces techniques pour améliorer la précision des prévisions de la demande. Dans le chapitre 3, nous proposons un modèle innovant de prédiction de la demande spécifiquement conçu pour les chaînes logistiques. Nous détaillons l'implémentation de nos modèles de machine learning, ainsi que les outils logiciels et bibliothèques utilisés. Nous évaluons également la performance de notre modèle en utilisant des métriques appropriées et discutons des résultats obtenus, ainsi que de leurs implications pratiques. Nous terminons ensuite par une synthèse générale et discutons des futures perspectives de recherche.

Généralités sur l'optimisation de la chaîne logistique

1.1 Introduction

La chaîne logistique joue un rôle essentiel pour la compétitivité et la performance économique des entreprises dans un contexte de marché globalisé. Elle ne se limite pas à la simple coordination des flux de biens et de services, mais englobe une série de processus complexes allant de l'approvisionnement à la distribution, en passant par la production et la vente.

Ce chapitre propose d'explorer les différentes facettes de la chaîne logistique. Nous commencerons par définir la chaîne logistique et décrire ses composantes, puis nous analyserons sa structure et les flux qui la traversent. Nous aborderons ensuite les processus essentiels qui sous-tendent son fonctionnement.

La gestion de la chaîne logistique sera également examinée en détail. Nous définirons ce concept, discuterons des tâches spécifiques assignées aux services SCM (Supply Chain Management), et identifierons les défis courants auxquels ils sont confrontés.

Enfin, nous nous focaliserons sur l'optimisation de la chaîne logistique. Nous présenterons les objectifs de cette optimisation, examinerons les méthodes traditionnelles et leurs limites, et mettrons en lumière l'apport de l'intelligence artificielle dans la résolution des défis logistiques. Cette analyse globale permettra de comprendre comment les entreprises peuvent améliorer leur réactivité, réduire les coûts et mieux satisfaire les attentes des clients.

1.2 Chaîne logistique

La chaîne logistique, comme illustré dans la figure 1.1, est un pilier fondamental pour les succès des opérations de toute entreprise. Dans cette section, nous présentons une vue d'ensemble de la chaîne logistique.

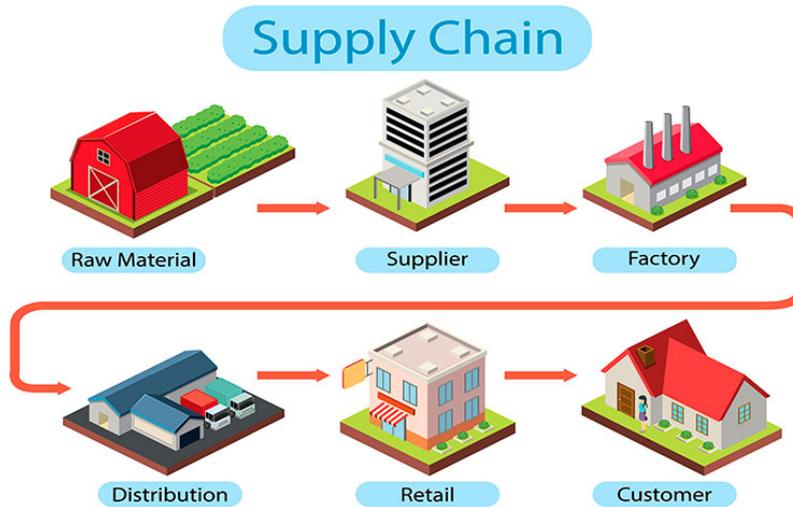


FIGURE 1.1 – Modèle de chaîne logistique [14]

1.2.1 Définition et composantes de la chaîne logistique

La chaîne logistique est un réseau complexe d'entités telles que des fournisseurs, des fabricants, des distributeurs et des détaillants. Elle exécute des fonctions essentielles telles que l'achat de matières premières, la fabrication de produits et la distribution de ces produits pour satisfaire la demande des clients. Le bon fonctionnement et l'efficacité d'une chaîne logistique sont essentiels pour une entreprise pour assurer des performances économiques satisfaisantes et rester compétitive sur le marché mondial [9]. Un exemple de chaîne logistique est présenté dans la figure 1.2.

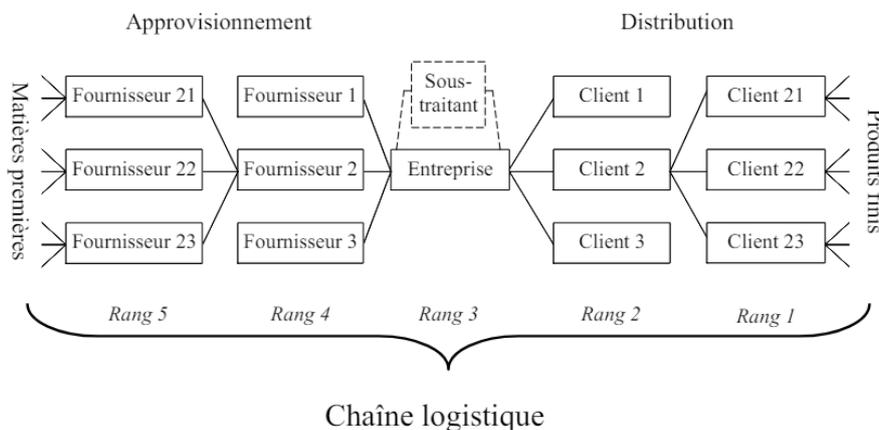


FIGURE 1.2 – Exemple de chaîne logistique [28]

Ce diagramme illustre une chaîne logistique structurée autour de deux principales composantes : l'approvisionnement et la distribution, chacune détaillant les éléments clés suivants :

* **Approvisionnement** :

- **Matières premières** : Fournies par trois niveaux de fournisseurs distincts (Fournisseur 1, Fournisseur 2 et Fournisseur 3).
- **Entreprise** : Au centre du diagramme, recevant les matières premières des fournisseurs.
- **Sous-traitant** : Connecté à l'entreprise par des liens en pointillés.

* **Distribution** :

- **Produits finis** : Fournis par l'entreprise à trois clients distincts (Client 1, Client 2 et Client 3).
- **Rangs** : Les différents niveaux de chaque élément sont indiqués, à savoir Rang 1 à Rang 5.

Les détails spécifiques concernent les flux entrants vers Fournisseur 2, représentant les matières premières fournies par Fournisseur 22. Ces matières premières sont utilisées par Fournisseur 2 pour produire des produits finis ou des composants qui seront ensuite acheminés vers l'Entreprise, soulignant ainsi l'alimentation directe par Fournisseur 22 à Fournisseur 2 dans la chaîne logistique.

Les niveaux spécifiques dans le diagramme sont organisés comme suit :

- **Rang 1** : Situé sous la section "Distribution", représente l'Entreprise comme le niveau le plus élevé dans la chaîne logistique.
- **Rang 2** : Également sous "Distribution", inclut les fournisseurs directs tels que Fournisseur 1, Fournisseur 2 et Fournisseur 3.
- **Rang 3** : Placé sous "Entreprise", représentant les sous-traitants.
- **Rang 4 et Rang 5** : Sous la section "Approvisionnement", correspondent aux niveaux inférieurs des fournisseurs.

1.2.2 Structure de la chaîne logistique

Il est important de définir une structure permettant de caractériser les différentes entités impliquées pour former une chaîne logistique. Dans la littérature scientifique, on distingue un ensemble de typologies usuelles, sur lesquelles sont fondées les modélisations existantes. Ces structures sont souvent classées en séries, dyades, divergences, convergences et réseaux. Les schémas fondamentaux sont exposés dans la figure 1.3 ci-dessous.

- **La structure série** : elle correspond à un processus de fabrication linéaire et vertical. Cette structure est pour étudier l'influence de la propagation de l'information sur l'ensemble de la chaîne [44].
- **La structure dyadique** : elle est considérée comme une variante spécifique d'une chaîne logistique en série, avec seulement deux niveaux. Elle constitue une référence pour l'analyse des relations entre

le client et le fournisseur, ou donneur d'ordre et le sous-traitant [44].

- **La structure divergente** : Elle permet de modéliser un réseau de distribution dans lequel les matières premières partent d'un point central et se répandent à travers toute la chaîne [44].
- **La structure convergente** : Elle permet de modéliser un processus d'assemblage. Dans une chaîne convergente, les matériaux transitent entre différents sites pour converger vers un point unique, généralement le site final d'assemblage [44].
- **La structure réseau** : c'est une combinaison des structures précédentes, intégrant à la fois les aspects d'approvisionnement et de distribution. Elle peut être plus complexe en raison du nombre accru d'acteurs impliqués, notamment pour les produits complexes [44].

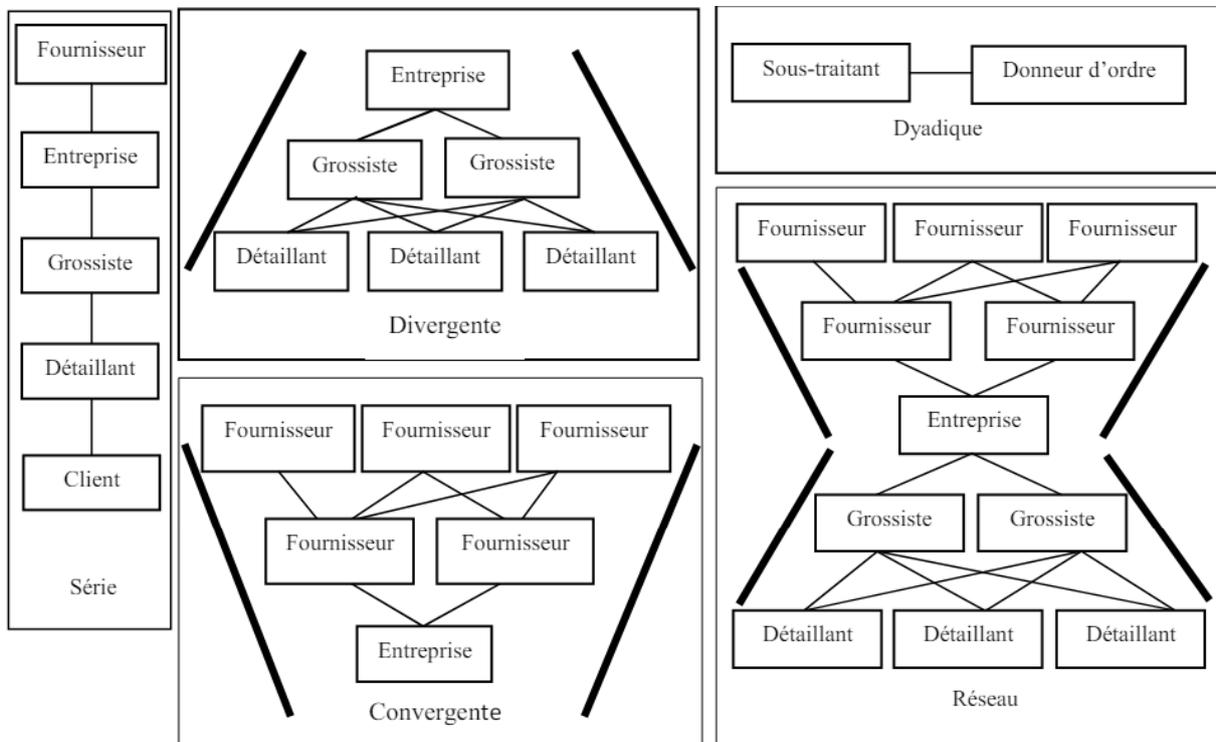


FIGURE 1.3 – Différentes structures de la chaîne logistique [44]

1.2.3 Les flux de la chaîne logistique

Nous décrivons ici les trois flux traversant une chaîne logistique illustrés par la figure 1.4 : le flux information, le flux physique et le flux financier.

- **Le flux d'information** :

Le flux d'information dans une chaîne logistique comprend les échanges de données entre les différents acteurs, incluant les commandes, les détails de livraison, et les informations techniques comme les capacités de production et les niveaux de stock [28].

- **Le flux physique :**

le flux physique est constitué par le mouvement des marchandises transportées et transformées depuis les matières premières jusqu'aux produits finis, en passant par les produits semi-finis. Ce flux justifie l'organisation d'un réseau logistique, incluant les sites de production, les moyens de transport et les espaces de stockage pour gérer les aléas [28].

- **Le flux financier :**

Le flux financier concerne la gestion financière des entreprises, incluant les ventes, les achats de matières premières et d'équipements, la location d'entrepôts et les salaires des employés. Il est généralement centralisé dans le service financier ou comptabilité, en lien avec les services achats et commercial [28].

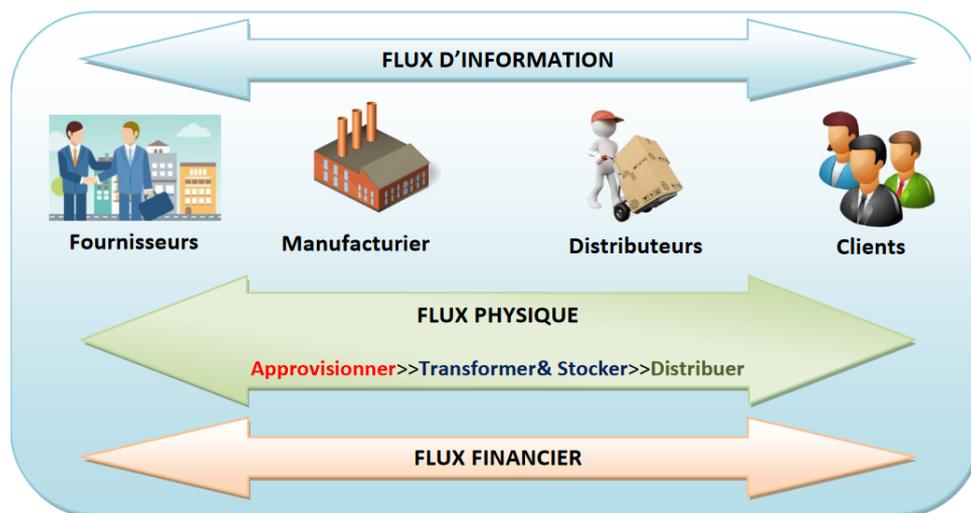


FIGURE 1.4 – Les Flux de la chaîne logistique[6]

1.2.4 Les processus de la chaîne logistique

Dans cette section, nous examinons les quatre processus essentiels de la chaîne logistique qui collaborent étroitement afin de répondre aux besoins des clients et de garantir leur satisfaction.

- **Approvisionnement** : le processus d'approvisionnement vise à fournir tous les composants nécessaires à la fabrication. On peut distinguer deux grandes phases principales : la sélection des fournisseurs et la passation des commandes. La première nécessite une évaluation sur différents critères tels que la qualité, le prix, les délais de réapprovisionnement, la capacité de production, la flexibilité à répondre à une demande variable et la possibilité d'effectuer des améliorations techniques sur les composants. La deuxième consiste à passer les commandes des composants nécessaires en fonction des besoins de production et à vérifier que la livraison comporte les bons composants, de qualité requise, en quantité conforme et au bon moment [28].

- **Production** :le processus de production concerne toutes les transformations que vont subir les composants pour réaliser les produits finis.Son objectif est de fabriquer les produits requis tout en assurant la productivité du système [28].
- **Distribution** :Le processus Distribution concerne la livraison des produits finis aux clients et reprend les questions d'optimisation des réseaux de distribution : l'organisation et le choix des moyens de transport, le choix du nombre d'étage dans le réseau de distribution ainsi que le positionnement des entrepôts et leur mode de gestion [28].
- **Vente** : le processus de vente géré par le service commercial, vise à entretenir des relations avec les clients et à mieux comprendre le marché. Il est également responsable de prévoir la demande et d'intégrer des aspects commerciaux tels que la durée de vie du produit pour anticiper ses ventes futures [28].

1.3 La gestion de la chaîne logistique (SCM : Supply Chain Management)

1.3.1 Définition de gestion de la chaîne logistique

La gestion de la chaîne logistique, comme illustré dans la figure 1.5, implique l'utilisation d'un ensemble d'approches pour intégrer efficacement les fournisseurs, les producteurs et les distributeurs. Son objectif est de garantir que les marchandises sont produites et livrées dans les quantités appropriées, aux emplacements adéquats et au moment idéal, tout en minimisant les coûts et en répondant aux exigences de service du client [6].



FIGURE 1.5 – Supply Chain Management : Modèle illustratif[13].

1.3.2 Les tâches dédiées au service SCM

Les activités relevant de la logistique couramment citées comprennent la gestion des stocks, l'entreposage, le transport, la distribution, l'import/export et l'optimisation de la chaîne logistique.

- **Gestion des stocks et de l'entreposage :**

La gestion des stocks est essentielle pour répondre à la demande, qu'elle provienne des clients finaux avec les produits finis, des besoins de production en matières premières et articles consommables, ou encore des pièces de rechange pour le service après-vente. La précision de la prévision de la demande est essentielle pour ajuster efficacement les niveaux de stock, minimisant ainsi les coûts tout en assurant la satisfaction des clients. En intégrant des techniques de prévision basées sur les données historiques et les modèles statistiques, les entreprises peuvent optimiser leurs opérations, réduire les risques de surstockage ou de rupture, et améliorer leur réactivité face aux fluctuations du marché [6].

- **Distribution**

La distribution comprend l'ensemble des opérations qui visent à mettre un produit ou service à la disposition du consommateur ou de l'utilisateur final, tout au long de la chaîne logistique. Cela

intervient après la production, de la fabrication ou de l'importation depuis la commercialisation par le producteur, fabricant ou importateur, jusqu'à la livraison au consommateur ou utilisateur final [6].

- **Transport**

La logistique du transport concerne principalement la gestion des flux de marchandises. Les principaux aspects de la chaîne du transport des marchandises figurent parmi les activités de transport local, national et international selon les différents modes de transport : terrestre, ferroviaire, maritime, aérien ou intermodal [6].

- **Import /Export**

Le service d'import/export est essentiel pour appliquer les règles douanières spécifiques à chaque pays. Chaque secteur possède ses propres spécificités pour réguler ces particularités. Les fonctions de base sont toute fois identiques et reprennent pour les échanges extra-nationaux les activités gérant les échanges «standards ». Les services d'import/export aident également les départements d'achat à choisir les partenaires commerciaux appropriés (transporteurs, transitaires, etc.) et les incoterms les plus adaptés pour chaque commande spécifique [6].

- **Optimisation de la chaîne logistique**

L'optimisation de la chaîne logistique vise à améliorer les performances en maximisant l'utilisation efficace des ressources internes et externes, assurant ainsi une coordination optimale à travers toute la chaîne. Cette approche transforme la compétition entre entreprises en une compétition entre chaînes logistiques, où la performance est déterminée par le rendement global de tous les acteurs économiques impliqués, y compris fournisseurs et acheteurs [21]. Plus de détails sur l'optimisation de la SCM seront présentés dans la section suivante.

1.3.3 Les défis courants rencontrés dans la gestion de la chaîne logistique

La gestion de la chaîne logistique est confrontée à divers défis qui peuvent limiter son efficacité et sa performance globale. Parmi ces défis figurent une gestion insatisfaisante des relations au sein de la chaîne logistique et une planification des ressources insuffisante entraînant des déséquilibres entre l'offre et la demande. En particulier, la gestion des stocks et la prévision de la demande posent des problèmes majeurs. Une gestion inadéquate des stocks peut entraîner des niveaux insuffisants ou excessifs, affectant la capacité à répondre rapidement aux besoins des clients tout en minimisant les coûts de stockage. De même, une prévision inexacte de la demande peut entraîner des fluctuations imprévues, entraînant des ruptures de stock ou des surplus inutiles. Ces deux aspects sont essentiels pour maintenir l'équilibre entre l'offre et la demande et pour assurer la performance optimale de la chaîne logistique [34]. Mettre en

lumière la prévision de la demande soulève les défis suivants [42] :

- **Saisonnalité** : La fluctuation des ventes en fonction des saisons nécessite des modèles capables de capturer ces variations pour éviter les surplus ou les ruptures de stock.
- **Cycles de vie des produits** : Les prévisions doivent tenir compte des phases de vie des produits, y compris le lancement, la croissance, la maturité et le déclin, pour ajuster les niveaux de stock en conséquence.
- **Volatilité de la demande clients** : Les changements rapides dans les comportements d'achat des consommateurs peuvent rendre difficile la prévision précise de la demande à court terme.
- **Promotions et événements** : Les promotions spéciales et les événements peuvent entraîner des pics de demande imprévus, nécessitant une adaptation rapide des prévisions pour répondre à ces fluctuations.
- **Commandes clients confirmées** : Intégrer les commandes clients confirmées dans les prévisions pour ajuster les niveaux de stock et éviter les ruptures de stock tout en minimisant les surplus.

Ces défis soulignent l'importance d'utiliser des méthodes de prévision avancées qui intègrent des données historiques et actuelles, ainsi que des techniques d'intelligence artificielle et d'apprentissage automatique pour améliorer la précision des prévisions et optimiser la gestion des stocks, ce qui sera discuté plus en détail dans la suite de ce mémoire.

1.4 L'optimisation de la chaîne logistique

Dans cette section, nous présentons les objectifs de l'optimisation d'une chaîne logistique, les limites d'optimisation avec les techniques traditionnelles et comment l'intelligence artificielle contribue à surmonter ces limites.

1.4.1 Objectifs de l'optimisation de la chaîne logistique

L'optimisation de la chaîne logistique vise à améliorer l'efficacité et la performance opérationnelle en minimisant les coûts et en maximisant la satisfaction des clients. Elle cherche à équilibrer l'offre et la demande, réduire les inventaires excédentaires, et augmenter la flexibilité et la réactivité face aux changements du marché [27]. Elle implique la coordination et l'optimisation de toutes les activités de la chaîne d'approvisionnement, depuis l'approvisionnement en matières premières jusqu'à la livraison des produits finis aux clients. Pour illustrer les objectifs atteints par l'optimisation de la chaîne logistique, voici quelques exemples concrets [20] :

- Réduire les coûts de transport : En optimisant les itinéraires de livraison et en consolidant les cargaisons, l'optimisation de la chaîne logistique peut réduire les coûts de transport de manière significative.

- Diminuer les stocks : En améliorant la prévision de la demande et en optimisant les niveaux de stock, l'optimisation de la chaîne logistique permet de réduire significativement les coûts de stockage et le risque de ruptures de stock. Une prévision de la demande plus précise permet aux entreprises de mieux anticiper les besoins futurs, évitant ainsi les surstocks coûteux et les pénuries qui peuvent entraîner des interruptions de production et des pertes de ventes.
- Accélérer les délais de livraison : En optimisant les processus de commande, d'entreposage et de livraison, l'optimisation de la chaîne logistique peut réduire les délais de livraison et améliorer la satisfaction des clients.
- Améliorer la visibilité de la chaîne d'approvisionnement : En mettant en place des systèmes de suivi et de traçabilité, l'optimisation de la chaîne logistique peut améliorer la visibilité de la chaîne d'approvisionnement et faciliter la prise de décision rapide.

1.4.2 Méthodes traditionnelles d'optimisation de la chaîne logistique et leurs limites

Les méthodes traditionnelles d'optimisation de la chaîne logistique se réfèrent aux techniques et aux approches utilisées avant l'avènement des technologies avancées telles que le ML et L'IA . Cela peut inclure des méthodes telles que les modèles mathématiques, les algorithmes heuristiques et les analyses statistiques [25]. Pour expliquer les méthodes et les limites des approches traditionnelles dans l'optimisation de la chaîne d'approvisionnement, voici quelques exemples [20] :

- Méthodes basées sur des modèles mathématiques : Les modèles mathématiques, tels que les modèles de programmation linéaire et les modèles de théorie des files d'attente, sont utilisés pour optimiser des aspects spécifiques de la chaîne d'approvisionnement comme l'allocation des ressources, la planification de la production et la gestion des stocks.
- Utilisation d'algorithmes heuristiques : Les algorithmes heuristiques, tels que ceux basés sur les algorithmes génétiques ou le recuit simulé (simulated annealing en anglais) qui est une technique utilisée pour trouver une solution approximative à un problème d'optimisation global, permettent de trouver des solutions approximatives à des problèmes complexes d'optimisation de la chaîne d'approvisionnement.
- Analyse statistique : Les méthodes d'analyse statistique sont employées pour examiner les données de la chaîne d'approvisionnement, identifier des tendances et des modèles afin d'améliorer l'efficacité opérationnelle.

Limites des méthodes traditionnelles :

- Complexité d'implémentation et de maintenance.

- Adaptabilité limitée aux changements dynamiques de la chaîne d'approvisionnement.
- Sous-exploitation des vastes quantités de données disponibles dans la chaîne d'approvisionnement moderne. Par exemple, une entreprise collecte des données détaillées sur les tendances d'achat et les performances des produits mais n'utilise que des analyses basiques comme le suivi des niveaux de stock, manquant ainsi l'opportunité d'appliquer des techniques avancées d'analyse prédictive pour anticiper les fluctuations de la demande et optimiser les niveaux de réapprovisionnement.

1.4.3 Contribution de l'IA à la résolution des défis de la chaîne logistique

L'intelligence artificielle (IA) a permis aux entreprises d'améliorer leurs prévisions, d'assurer une visibilité en temps réel et de faciliter la prise de décision tout au long de la chaîne logistique. Les algorithmes d'IA dédiés à l'analyse prédictive analysent de grandes quantités de données historiques, et produisent des prévisions plus précises que les méthodes traditionnelles. Ils peuvent prédire la demande future, optimiser les processus opérationnels et minimiser les coûts liés aux stocks. De plus, les technologies d'IA permettent une visibilité en temps réel dans les opérations de la chaîne logistique, facilitant la surveillance des flux de marchandises et la gestion des perturbations. Enfin, les systèmes d'aide à la décision facilitent la prise de décisions éclairées en interagissant avec des environnements et en analysant des données complexes pour recommander des décisions avisées [31].

1.5 Conclusion

Dans ce chapitre, nous avons exploré les composantes clés et les processus essentiels de la chaîne logistique, soulignant son rôle dans la performance des entreprises. Nous avons également discuté des défis courants et de l'importance de l'optimisation, en montrant comment l'intelligence artificielle et les technologies avancées transforment cette gestion. En particulier, nous avons mis en lumière la gestion des stocks et la prévision de la demande comme éléments critiques pour l'efficacité et la réactivité de la chaîne logistique. Dans le chapitre suivant, nous détaillerons les méthodes spécifiques d'optimisation basées sur l'IA et expliquerons leur fonctionnement pour améliorer la chaîne logistique.

Etat de l'art sur l'IA et la prévision de la demande dans une chaîne logistique

2.1 Introduction

Les modèles d'apprentissage automatique jouent un rôle essentiel dans la prévision de la demande au sein de la chaîne logistique. L'objectif principal est de parvenir à une prédiction des demandes futures en se basant sur les données historiques disponibles. Ce chapitre examinera la littérature la plus récente dans ce domaine en identifiant les modèles de prédiction et les types de données les plus couramment utilisés, ainsi que les métriques d'évaluation.

2.2 Prévision de la demande pour la gestion de la chaîne logistique

Des prévisions précises sont essentielles pour toute planification commerciale. D'une part des estimations excessivement élevées entraînent une surproduction, augmentant ainsi les coûts liés aux ressources telles que les matières premières, la main-d'œuvre et l'espace de stockage. D'autre part une sous-production peut entraîner des pénuries, des opportunités commerciales manquées, et une potentielle perte de clients insatisfaits.

La prévision de la demande comme illustré dans la figure 2.1 joue un rôle essentiel au sein de la chaîne logistique. C'est une clé indispensable pour la planification et la fixation des objectifs de chaque entreprise. En fournissant des informations critiques, elle améliore la gestion des stocks et permet une

adaptation optimale de l'offre à la demande. Grâce à ces prévisions, les entreprises peuvent non seulement optimiser leurs ressources mais aussi augmenter leur réactivité et leur flexibilité face aux changements du marché [39].

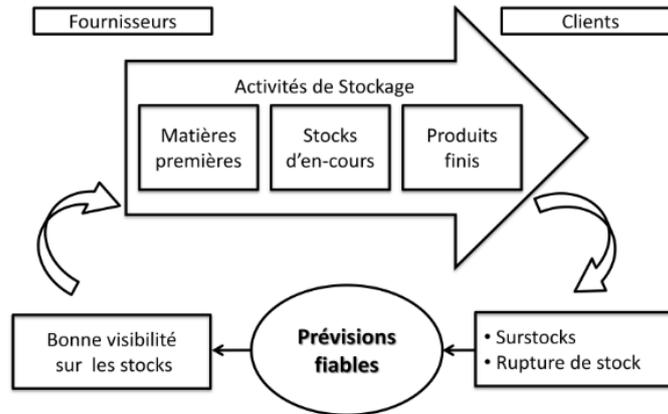


FIGURE 2.1 – Prévision de la demande et gestion des stocks [39]

2.2.1 Rôle de la prévision de la demande

Pour mieux comprendre l'impact de la prévision de la demande sur la chaîne logistique, examinons quelques points clés :

- **Amélioration de la Précision et de l'Efficacité** : La prévision de la demande permet d'anticiper les besoins des clients et de planifier efficacement les opérations de la chaîne logistique, y compris la gestion des stocks, la production et la distribution. Cela réduit les coûts opérationnels et améliore la satisfaction des clients [26].
- **Réduction des Coûts** : En fournissant des estimations précises de la demande future, les entreprises peuvent éviter les surstocks et les ruptures de stock, réduisant ainsi les coûts de stockage et de pénurie. Cela aide également à optimiser les niveaux de production et à ajuster les stratégies d'approvisionnement [26].
- **Optimisation des Ressources** : La prévision de la demande aide à allouer efficacement les ressources, telles que la main-d'œuvre, les matières premières et les capacités de production. Cela assure une utilisation optimale des ressources, améliorant ainsi la productivité globale de la chaîne logistique [26].
- **Réponse aux Fluctuations du Marché** : La prévision de la demande permet aux entreprises de réagir rapidement aux changements du marché et aux comportements des consommateurs, ce qui est crucial dans des environnements dynamiques où les préférences des clients et les conditions du marché peuvent changer rapidement [26].

- **Planification à Long Terme** : Une bonne prévision de la demande permet une meilleure planification stratégique à long terme. Elle aide les entreprises à identifier les tendances à long terme et à adapter leurs stratégies de développement de produits, d'expansion de marché et d'innovation [26].

2.2.2 L'Intelligence Artificielle (IA) dans la prévision de la demande

L'IA révolutionne la prévision de la demande en fournissant des solutions efficaces pour anticiper les besoins des clients. En analysant des quantités massives de données, l'IA peut offrir des informations décisionnelles précises, permettant aux entreprises d'ajuster leurs stocks et leur production en conséquence. Dans le secteur du commerce de détail, cela se traduit par des prévisions plus précises, une gestion optimisée des stocks, et une réduction des coûts [37].

La prévision de la demande par l'IA, qui analyse des ensembles de données plus vastes et diversifiés, a rapidement gagné en popularité, promettant des prédictions plus précises des achats des clients. Selon Gartner [43], c'est l'application la plus couramment utilisée de l'IA dans la planification de la chaîne d'approvisionnement, adoptée par 45 % des entreprises mondiales. McKinsey [43] prévoit que ces prévisions par l'IA pourraient réduire les erreurs de 30 à 50 %.

2.3 Modèles de prédiction de la demande

Dans cette section, nous présentons les modèles de Machine Learning conventionnels les plus couramment utilisés dans la prédiction de la demande illustré dans la figure 2.2. Les chercheurs, experts et praticiens choisissent celui qui convient le mieux à leurs besoins, en fonction de leur expériences et de leurs objectifs spécifiques en matière de ML. Les modèles de prédiction de la demande diffèrent en fonction de leur complexité, des modèles simples, modèles d'ensemble et des modèles hybrides.

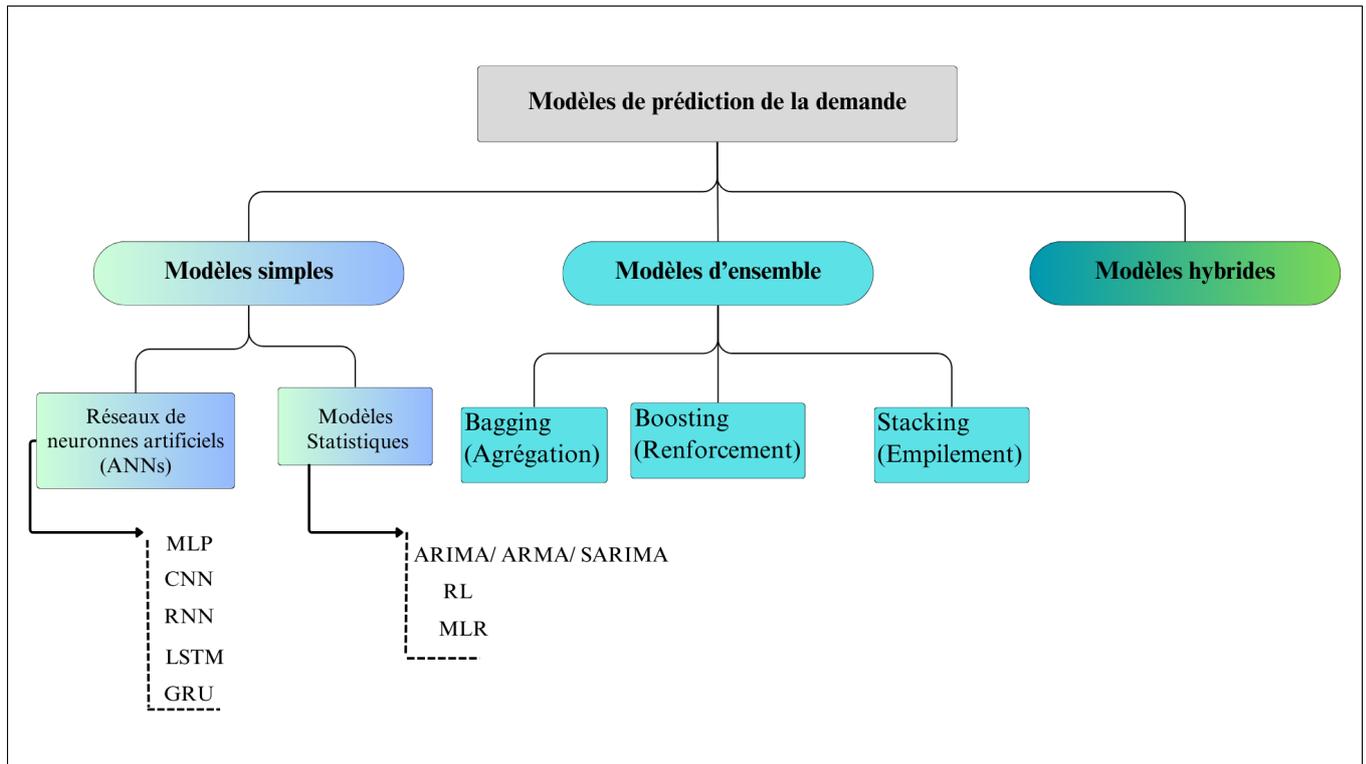


FIGURE 2.2 – Modèles de prédiction de la demande

2.3.1 Modèles simples

Un modèle simple en apprentissage automatique désigne un modèle spécifique conçu pour résoudre une tâche ou un problème particulier en utilisant un seul algorithme ou une seule architecture. Exemple de ces modèles :

- **Modèles statistiques**

Ces modèles utilisent des méthodes statistiques pour analyser et prédire les données. Parmi les modèles statistiques couramment utilisés, on trouve :

- ARMA/ARIMA/SARIMA : sont des outils essentiels pour analyser et prévoir les séries temporelles. ARMA (Autoregressive Moving Average) combine la régression autoregressive avec la moyenne mobile, ARIMA (Autoregressive Integrated Moving Average) étend cette approche en intégrant une composante d'intégration pour traiter les tendances, tandis que SARIMA (Seasonal Autoregressive Integrated Moving Average) incorpore également des effets saisonniers. Ces modèles permettent de capturer les tendances, les cycles et les variations saisonnières dans les données, offrant ainsi des perspectives précieuses pour la modélisation et la prévision dans divers domaines [18].

- LR (Régression Linéaire) : elle étudie la relation entre une variable dépendante et une ou plusieurs variables indépendantes, cherchant à modéliser cette relation par une ligne droite et à estimer les coefficients associés [18].
- MLR (Régression Linéaire Multiple) : elle étend la régression linéaire simple pour modéliser la relation entre une variable dépendante et plusieurs variables indépendantes simultanément, permettant ainsi de prendre en compte plusieurs facteurs dans le modèle ce qui améliore la précision des prédictions [18].

- **Réseaux de neurones artificiels (ANN)**

Les réseaux de neurones artificiels (ANN) sont des algorithmes inspirés du cerveau humain, utilisés pour traiter des données complexes et effectuer des prédictions. Ils sont composés de couches de neurones interconnectés. Les ANN comprennent généralement une couche d'entrée pour recevoir les données initiales, une ou plusieurs couches cachées pour effectuer des calculs complexes, et une couche de sortie pour produire les résultats [5]. Cette architecture est illustrée par la figure 2.3 qui montre l'organisation typique des couches et des connexions dans un ANN. Certains des types d'ANN les plus courants incluent :

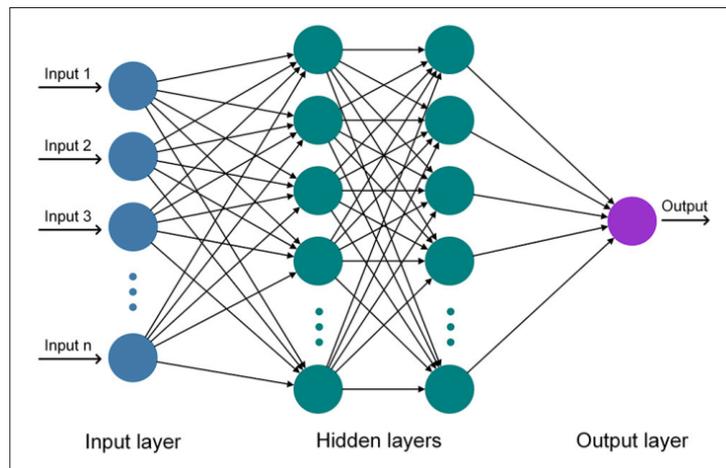


FIGURE 2.3 – Architecture d'un Réseau de Neurones Artificiels (ANN)

- **MLP (Multi-Layer Perceptron) :**

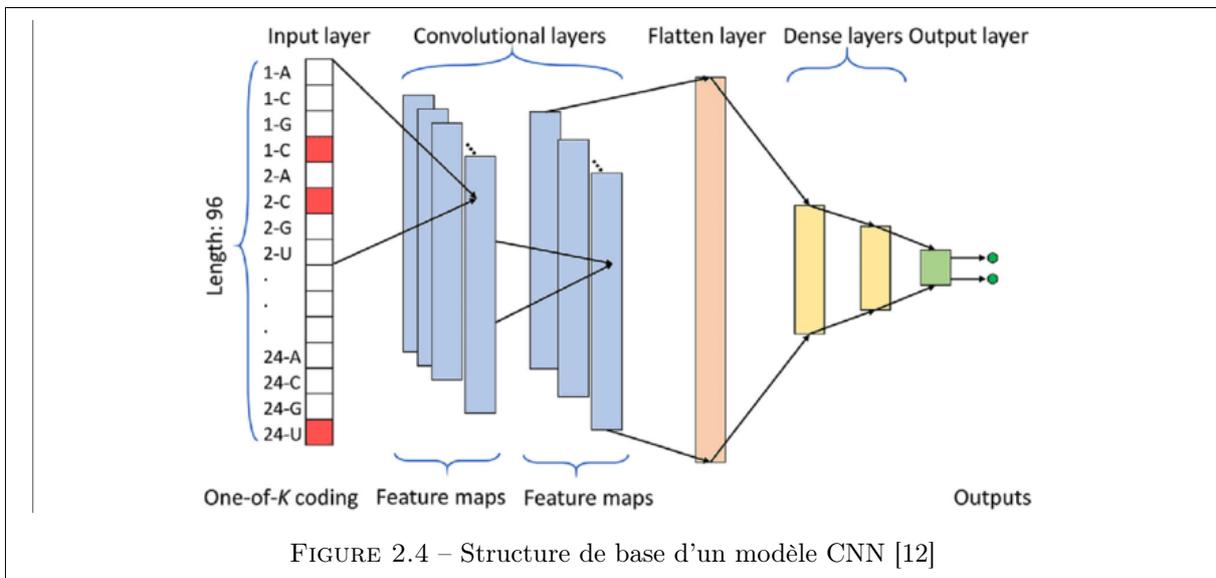
Le Multi-Layer Perceptron (MLP) est un type fondamental de réseau de neurones artificiels (ANNs) qui opère sur un modèle de propagation avant. Composé de plusieurs couches entièrement connectées, il est spécialement conçu pour traiter des données séquentielles telles que les séries temporelles [2]. Le MLP tire parti de sa capacité à découvrir et à modéliser des relations complexes entre les caractéristiques d'entrée et la variable cible. En utilisant des fonctions d'activation non linéaires, il extrait des caractéristiques globales des données, permettant ainsi une représentation hiérarchique et abstraite des informations [41].

○ **1D-CNN(Convolutional Neural Network unidimensionnel)** :

Les réseaux de neurones convolutifs (CNN) sont initialement conçus pour l'analyse d'images en raison de leur capacité à extraire des relations locales invariantes dans les dimensions spatiales des données. Cette architecture utilise des filtres convolutifs pour détecter des motifs spécifiques dans une image, ce qui en fait un outil puissant pour la reconnaissance d'objets et d'autres tâches liées à la vision par ordinateur [2].

Les réseaux de neurones convolutifs unidimensionnels (1D-CNN) illustrés dans la figure 2.4 sont une extension de cette architecture, spécifiquement adaptée pour traiter des données séquentielles telles que les séries temporelles. Contrairement aux CNN traditionnels qui opèrent sur des images 2D, les 1D-CNN appliquent des convolutions le long de la dimension temporelle des données. Cela leur permet d'identifier des motifs locaux et d'extraire des caractéristiques pertinentes pour la prédiction ou l'analyse de données séquentielles, comme les séries chronologiques.

Ainsi, les 1D-CNN sont utilisés pour leur capacité à capturer des relations temporelles complexes et à intégrer la non-linéarité grâce aux fonctions d'activation, ce qui les rend particulièrement efficaces dans divers domaines, notamment la prédiction de la demande [41].



○ **RNN(Recurrent Neural Networks)** :

Les Réseaux de Neurones Récurrents (RNNs) sont une catégorie de réseaux neuronaux spécialement conçus pour modéliser des données séquentielles et effectuer des prédictions basées sur des informations historiques dans une séquence donnée. Contrairement aux réseaux neuronaux traditionnels avec des données de taille fixe, les RNNs peuvent traiter des séries temporelles de longueurs variables. Chaque neurone récurrent reçoit des entrées à chaque étape temporelle

et utilise sa propre sortie de l'étape précédente comme entrée, ce qui lui permet de maintenir une forme de mémoire et de capturer les dépendances temporelles.

Pour résoudre les défis tels que le problème de disparition ou d'explosion des gradients lors de l'entraînement sur de longues séquences, les RNNs utilisent des techniques comme la rétro-propagation tronquée dans le temps. Cependant, cela peut limiter leur capacité à apprendre des dépendances à très long terme. Pour remédier à cela, des architectures de cellules spéciales comme les LSTM (Long Short-term Memory) et les GRU (Gated Recurrent Unit) ont été développées. Ces modèles permettent aux RNNs de mieux gérer les informations sur une période prolongée, améliorant ainsi leurs performances dans la prédiction de séries chronologiques et d'autres applications nécessitant une modélisation temporelle [2].

- **LSTM (Long Short-Term Memory Network) :**

LSTMs sont une architecture de réseau neuronal récurrent, comme illustré dans la figure 2.5, conçue pour capturer les dépendances à long terme dans les données séquentielles. Proposés par Hochreiter et Schmidhuber en 1997 et améliorés au fil du temps, les LSTM surmontent les limitations des réseaux neuronaux récurrents traditionnels en utilisant des unités de mémoire et des portes pour réguler le flux d'informations. Chaque cellule LSTM dispose d'un état à court terme et d'un état à long terme, gérés par des portes d'entrée, de sortie et d'oubli.

Les portes d'entrée contrôlent les informations nouvelles à ajouter à l'état à long terme, tandis que les portes d'oubli décident des parties de cet état à effacer. En parallèle, la porte de sortie sélectionne les parties pertinentes de l'état à long terme pour produire la sortie actuelle. Ainsi, les LSTM peuvent apprendre à conserver des informations importantes sur des périodes prolongées, ce qui les rend efficaces pour la prédiction de séries chronologiques et d'autres applications nécessitant une mémoire à long terme des séquences [2].

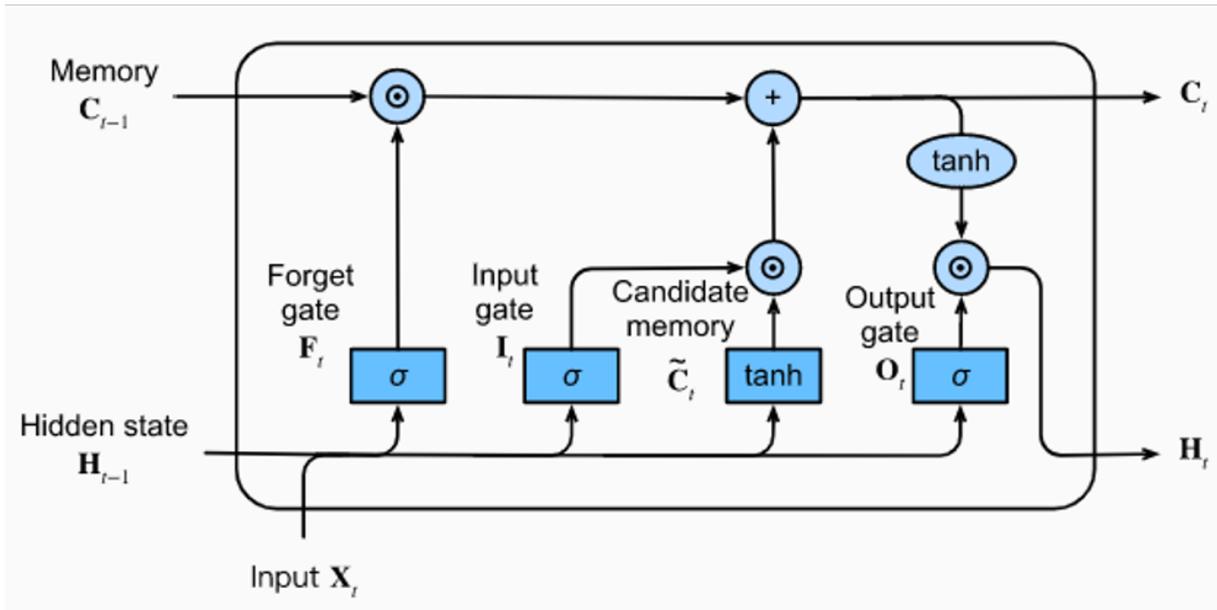


FIGURE 2.5 – Structure de base d'un modèle LSTM [22]

- **GRU (Gated Recurrent Unit) :**

Le modèle GRU, introduit par Cho et al. en 2014, illustré par la figure 2.6 représente une avancée significative dans le domaine des réseaux de neurones récurrents. Conçu comme une version simplifiée des LSTM de Hochreiter and Schmidhuber (1997), le GRU maintient une efficacité comparable tout en réduisant le nombre de paramètres requis. Son architecture fusionne les fonctionnalités des portes d'oubli et d'entrée des LSTM en une seule entité appelée la porte de mise à jour, simplifiant ainsi l'entraînement et l'interprétation du modèle tout en conservant des performances robustes dans les tâches séquentielles.

En termes de fonctionnement, la cellule GRU réorganise le modèle LSTM en fusionnant les états de mémoire et en simplifiant la gestion des portes de contrôle. Contrairement aux LSTM qui ont une porte de sortie distincte, le GRU utilise un mécanisme simplifié où un seul contrôleur de porte régle à la fois l'activation de la porte d'entrée et de la porte d'oubli. Cela permet au modèle de mieux gérer et de sélectionner les informations pertinentes pour la prédiction à chaque étape temporelle, tout en minimisant la complexité du réseau [2].

Le modèle GRU se distingue par sa capacité à capturer efficacement les dépendances à long terme dans les séquences, le rendant particulièrement adapté à la modélisation des tendances et de la saisonnalité dans les données de séries temporelles. Cette capacité est cruciale pour la prévision de la demande, où une précision élevée dans l'anticipation des variations périodiques, influencées par les saisons et les cycles annuels, est essentielle [32].

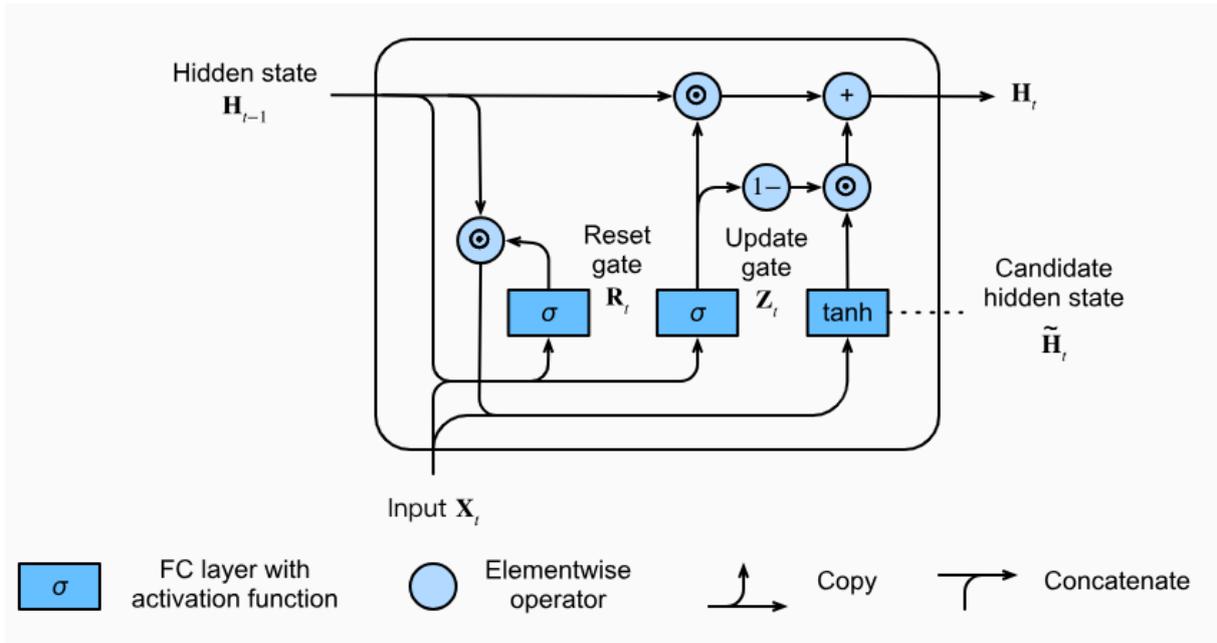


FIGURE 2.6 – Structure de base d'un modèle GRU. [1]

2.3.2 Modèles d'ensemble

Les modèles d'ensemble utilisent plusieurs modèles pour améliorer la précision des prévisions. En combinant plusieurs modèles, ces méthodes visent à réduire les erreurs de prévision en tirant parti des points forts de chaque modèle individuel [24]. La figure 2.7 présente une vue d'ensemble des principales méthodes d'ensemble utilisées en apprentissage automatique [15].

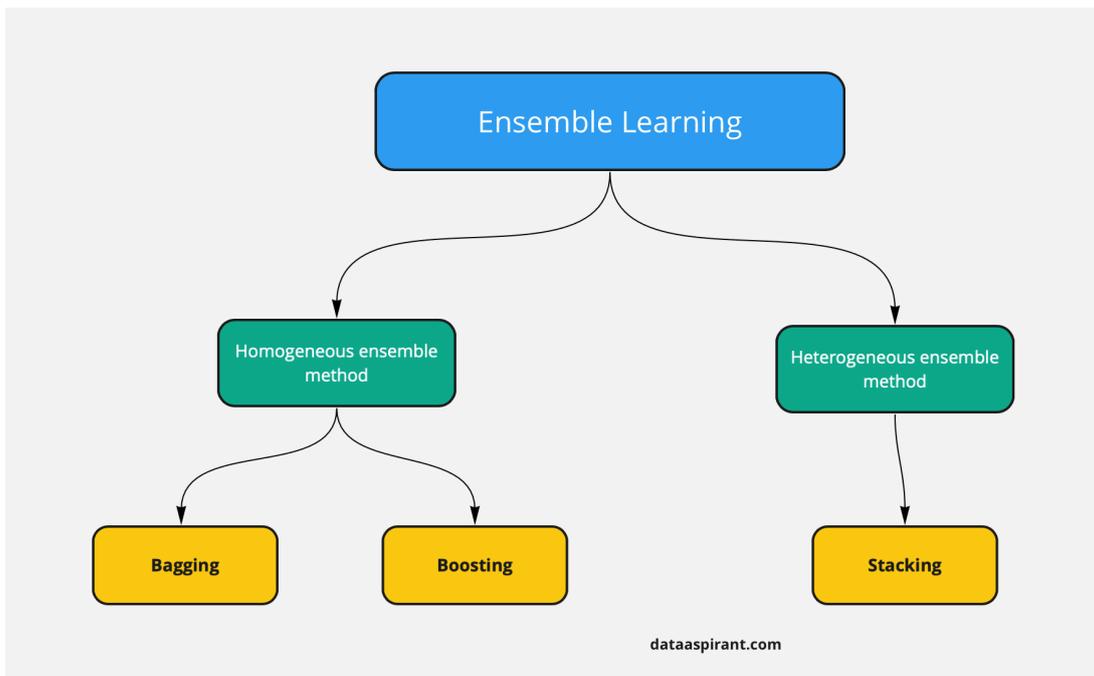


FIGURE 2.7 – Illustration des méthodes d'ensemble en apprentissage automatique [15]

Les principales techniques de ces modèles incluent :

- **Boosting** : Également appelé "Renforcement" en français, c'est une technique d'ensemble qui combine plusieurs modèles faibles pour créer un modèle fort. Chaque modèle successif essaie de corriger les erreurs commises par les modèles précédents, augmentant ainsi la précision globale de la prédiction [24]. Dans la figure 2.8 sont illustrées les étapes du Boosting.

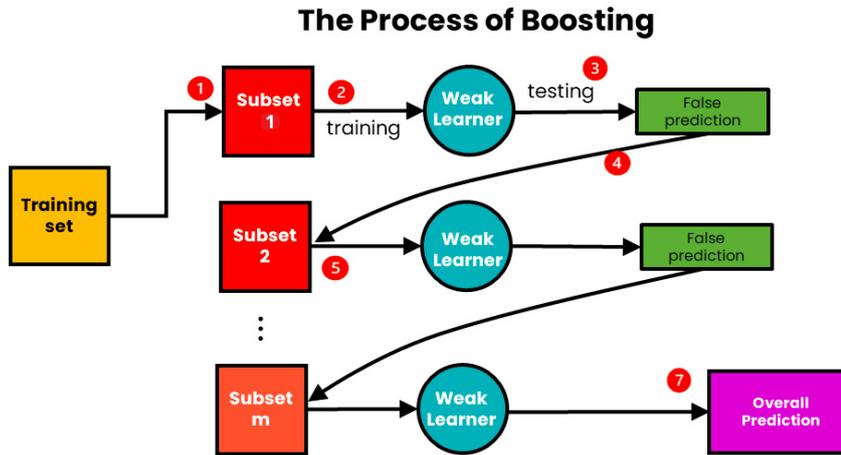


FIGURE 2.8 – Étapes du Boosting [3]

- **Bagging** : Connu sous le nom de Agrégation en français, appelé aussi bootstrap aggregating, est une méthode d'ensemble qui améliore la précision en générant plusieurs versions d'un prédicteur et en utilisant la moyenne de ces prédicteurs. Cette méthode réduit la variance et aide à éviter le surapprentissage [24]. Un exemple populaire de bagging est Random Forest. Dans la figure 2.9 sont illustrées les étapes du bagging.

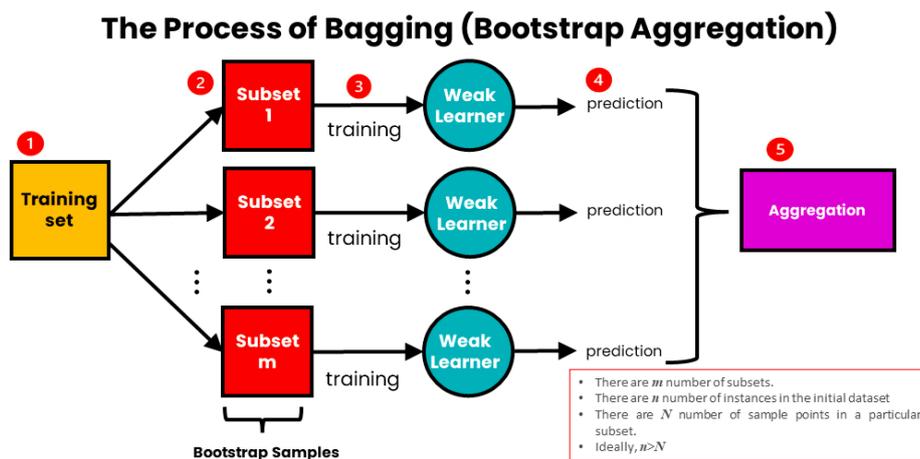


FIGURE 2.9 – Étapes du Bagging [3]

- **Stacking** : Le stacking, également appelé stacked generalization ou Empilement en français, est une

méthode d'ensemble qui combine plusieurs modèles de différents types. Les prédictions des modèles de base sont utilisées comme entrées pour un modèle méta, qui tente d'apprendre la meilleure combinaison de ces prédictions pour améliorer les performances globales [24]. Dans la figure 2.10 sont illustrées les étapes du Stacking.

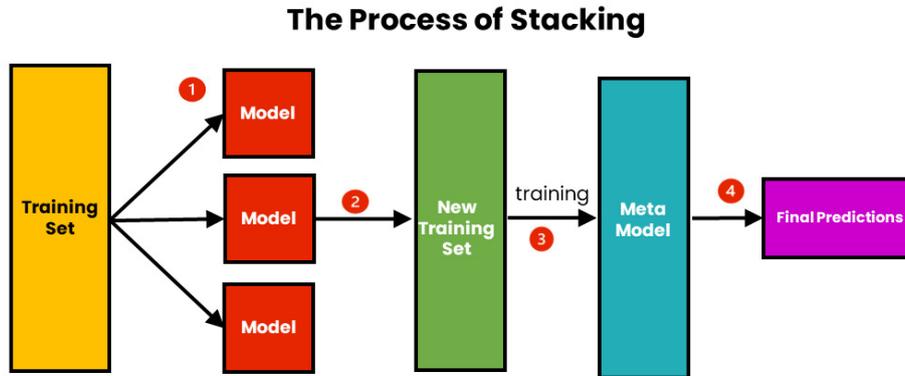


FIGURE 2.10 – Étapes du Stacking
[3]

Parmi les algorithmes d'apprentissage d'ensemble les plus populaires, on a : Random Forest, XGBoost et LightGBM.

- **Random Forest** : cet algorithme est une méthode puissante en machine learning. Il crée un ensemble d'arbres de décision lors de l'entraînement, chacun étant construit sur un sous-ensemble aléatoire des données. En agrégeant les prédictions de ces arbres, par vote pour la classification ou moyenne pour la régression, il offre des résultats précis et stables. Les Random Forest sont appréciés pour leur capacité à gérer des données complexes, réduire le surajustement et fournir des prévisions fiables dans divers contextes [17]. Il appartient à la méthode Bagging.
- **Xgboost** : une bibliothèque de boosting en machine learning qui agit comme un métamodèle pour combiner les prédictions de plusieurs modèles de base. Son objectif principal est d'améliorer la précision des prédictions en optimisant une fonction de perte spécifiée. XGBoost est largement utilisé pour résoudre des problèmes de régression et de classification, offrant une performance élevée tout en gérant efficacement le surapprentissage [33]. Il appartient à la méthode Boosting.
- **LightGBM** : Un algorithme de boosting rapide et efficace. Il est également défini comme une machine de gradient boost (boosting) qui se distingue par sa vitesse de traitement rapide et son efficacité, particulièrement adaptée au traitement de grands ensembles de données. Développé par Microsoft en 2017, LightGBM utilise une approche de croissance d'arbre basée sur les gradients et l'échantillonnage unilatéral pour filtrer efficacement les instances de données et les caractéristiques moins utiles pour améliorer le modèle [30]. Il appartient à la méthode Boosting.

2.3.3 Modèles hybrides

Un modèle hybride combine deux ou plusieurs techniques ou modèles distincts afin de créer un nouveau modèle intégrant les points forts de chacun. Cette approche vise à améliorer la performance prédictive en exploitant les attributs supérieurs de chaque composant et en palliant les limitations individuelles. Par exemple, on pourrait combiner des modèles statistiques avec des techniques de machine learning, ou fusionner des architectures neuronales [24]. Par exemple, utiliser les couches de CNN et GRU dans un seul modèle neural pour exploiter à la fois les capacités de traitement spatial et temporel des données. nous définissons le modèle du pseudo code suivant 1 qui combine des couches de CNN et de GRU dans une architecture neuronale unique.

Algorithm 1: Modèle hybride `model_cnn_gru`

Input: Données d'entrée `X_train_series_sub`**Output:** Prédiction

- 1 **Ajouter Couche** `TimeDistributed(Conv1D(filters=64, kernel_size=1, activation='relu'), input_shape=(None, X_train_series_sub.shape[2], X_train_series_sub.shape[3]))`
 - 2 **Ajouter Couche** `TimeDistributed(MaxPooling1D(pool_size=2))`
 - 3 **Ajouter Couche** `TimeDistributed(Flatten())`
 - 4 **Ajouter Couche** `GRU(50, activation='relu')`
 - 5 **Ajouter Couche** `Dense(1)`
 - 6 **Compiler** avec `loss='mse'` et `optimizer=adam`
 - 7 **Fin du modèle**
-

2.3.4 Différence entre les modèles hybrides et l'approche ensembliste

Les modèles hybrides et l'approche ensembliste se distinguent par leurs objectifs et leurs méthodes. Les modèles hybrides sont conçus pour intégrer différentes techniques ou architectures dans un seul modèle afin de combiner leurs forces et de pallier leurs limitations individuelles, visant ainsi à améliorer la performance prédictive en exploitant des capacités complémentaires. En revanche, l'approche ensembliste combine plusieurs modèles indépendants pour augmenter la robustesse et la précision des prédictions en agrégeant les résultats de chaque modèle, réduisant ainsi le risque d'erreur et de surapprentissage.

2.4 Les séries chronologiques

Les séries chronologiques représentent l'évolution d'un phénomène au fil du temps, elles se caractérisent par des observations prises à partir de la même source sur une période de temps fixe. L'objectif principal de l'étude des séries chronologiques est d'analyser et de prédire des observations futures [7]. Les données de séries temporelles peuvent être divisées en deux catégories :

- **Les séries temporelles univariées :**

Elles Concernent une seule variable étudiée sur une période de temps. Cela signifie qu'il n'y a qu'une colonne de données représentant une séquence de valeurs observées à différents moments. L'analyse de ces séries repose uniquement sur les valeurs présentes et passées de cette seule variable pour effectuer des prédictions.

- **Séries temporelles multivariées :**

Elles impliquent plusieurs variables observées simultanément sur une période de temps. Chaque colonne de données représente une séquence de valeurs pour une variable spécifique, et l'analyse prend en compte les interactions et les relations entre ces variables pour améliorer les prédictions.

Une série temporelle se compose généralement de trois composantes principales : **la tendance**, **la saisonnalité** et **les résidus**. La tendance décrit le mouvement à long ou court terme de la série temporelle indiquant ainsi une augmentation, une diminution ou une stagnation des valeurs au fil du temps. La saisonnalité représente un phénomène périodique qui se répète à intervalles réguliers, souvent influencées par des facteurs saisonniers ou cycliques, tandis que les résidus représentent des fluctuations aléatoires. Pour comprendre le comportement global de la série temporelle, il est nécessaire d'analyser toutes les composantes.

- **Analyse des séries temporelles avec les techniques traditionnelles et les modèles de Deep Learning**

Traditionnellement, l'analyse des séries temporelles nécessite la décomposition de la série en ses composantes pour faciliter l'interprétation et la prédiction. Les modèles statistiques classiques, tels que les modèles ARIMA, SARIMA et les modèles de lissage exponentiel, sont couramment utilisés pour capturer la tendance, la saisonnalité et les résidus [8].

En revanche, les modèles de deep learning sont capables d'extraire les informations pertinentes dans des données brutes sans nécessiter une décomposition de la série temporelle. Ils sont capables de gérer des relations complexes non linéaires, ce qui réduit le besoin d'ingénierie manuelle des caractéristiques et de concevoir des modèles. Les réseaux de neurones récurrents (RNNs) et leurs dérivés se sont montrés efficaces dans le traitement des séries temporelles [8].

2.5 Métriques d'évaluation pour les modèles de prédiction

Nous présentons ci-dessous les métriques les plus utilisées dans la revue de littérature :

- **Erreur moyenne absolue (Mean Absolute Error - MAE) :**

MAE est la mesure de la différence entre les valeurs prédites et les valeurs réelles. Elle est obtenue en calculant la moyenne des valeurs absolues de ces écarts, définie par la formule suivante :

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

Où :

— n est le nombre total d'observations.

- y_i représente la valeur réelle de la i -ème observation.
- \hat{y}_i représente la valeur prédite par le modèle pour la i -ème observation.

• **Erreur quadratique moyenne (Mean Squared Error - MSE) :**

MSE est la mesure de la différence entre les valeurs prédites et les valeurs réelles. Elle est calculée en prenant la moyenne des carrés de ces écarts, définie par la formule suivante :

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

- n est le nombre total d'observations.
- y_i représente la valeur réelle de la i -ème observation.
- \hat{y}_i représente la valeur prédite par le modèle pour la i -ème observation.

• **Erreur quadratique moyenne racine (Root Mean Squared Error - RMSE) :**

RMSE est la racine carrée de l'erreur quadratique moyenne. Elle est définie par la formule suivante :

$$\text{RMSE} = \sqrt{\text{MSE}} \quad (3)$$

• **Erreur absolue moyenne pondérée (Mean Absolute Percentage Error - MAPE) :**

MAPE est une mesure de l'erreur moyenne en pourcentage entre les valeurs prédites et les valeurs réelles d'un ensemble de données. Elle est calculée en prenant la moyenne de la valeur absolue de ces différences pondérées par les valeurs réelles, définie par la formule suivante :

$$\text{MAPE} = \frac{100}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \quad (4)$$

Où :

- y_i est la valeur réelle.
- \hat{y}_i est la valeur prédite.
- n est le nombre total d'échantillons.

• **Coefficient de détermination (R-squared) :** Le coefficient de détermination est une mesure de la proportion de variance dans les données expliquée par le modèle. Il est calculé en divisant la variance expliquée par la variance totale, définie par la formule suivante :

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (5)$$

où :

- y_i représente les valeurs observées.
- \hat{y}_i représente les valeurs prédites par le modèle.
- \bar{y} est la moyenne des valeurs observées.
- n est le nombre d'observations.

2.6 Travaux connexes

Dans cette section, nous passons en revue la littérature pertinente sur la prévision de la demande dans la chaîne logistique, en mettant l'accent sur les méthodes et les modèles utilisés, ainsi que sur les données employées et leurs résultats.

- **Mahya Seyedana, Fereshteh Mafakherib, et Chun Wang (2023) [41]** ont proposé une étude visant à prédire la demande future dans le domaine de la vente au détail en ligne en utilisant des méthodes de prévision basées sur l'apprentissage profond en ensemble, avec une comparaison de leurs performances par rapport à l'apprentissage à modèle unique. Leur approche combine les avantages des modèles individuels pour améliorer la précision des prévisions en exploitant les forces complémentaires de différents modèles.

La figure 2.11 présente l'approche proposée d'apprentissage profond en ensemble pour la prévision des séries chronologiques de la demande. L'objectif est d'établir des prévisions de la demande quotidienne en utilisant un réseau neuronal d'apprentissage profond en ensemble. Les auteurs détaillent la construction des modèles de base (MLP, LSTM et 1D-CNN), ainsi que l'intégration via un méta-apprenant, qui représentent les deux étapes clés de leur approche.

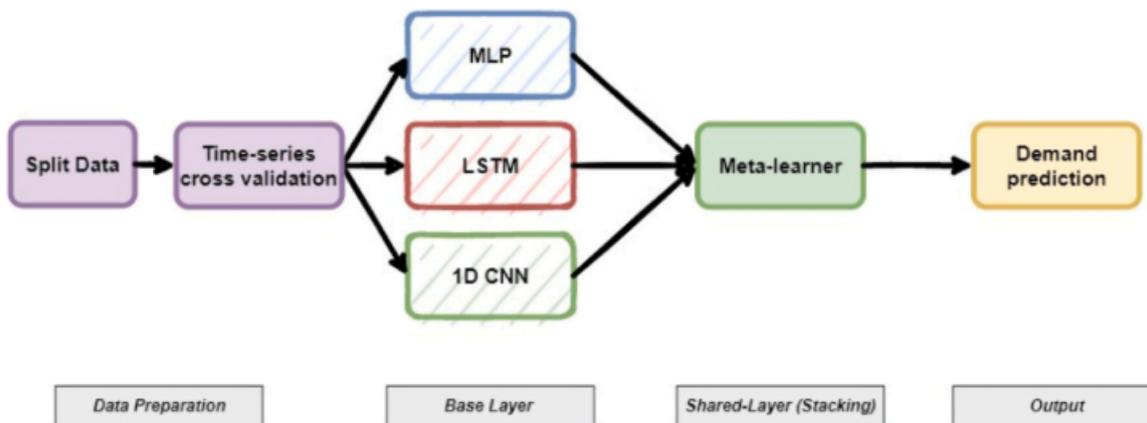


FIGURE 2.11 – Prévision de la demande en séries temporelles à l'aide d'un ensemble de deep learning [41]

Les chercheurs ont expérimenté leur approche sur deux ensembles de données de séries chronolo-

giques du monde réel dans la chaîne d'approvisionnement, comprenant des fournitures électroniques et sportives et couvrant la période de janvier 2015 à septembre 2017, pour démontrer l'efficacité et la praticité du modèle proposé .

Les résultats des métriques de performance pour les modèles de base (MLP, LSTM et 1D-CNN) ainsi que pour le méta-modèle sont présentés ci-dessous pour les deux ensembles de données. Les valeurs de l'erreur absolue moyenne en pourcentage (MAPE), de l'erreur absolue moyenne (MAE) et de l'erreur quadratique moyenne (RMSE) ont été utilisées pour évaluer la précision des prédictions.

Pour le premier ensemble de données (fournitures sportives) :

- Le MLP a obtenu une MAPE de 13.03% , une MAE de 125.90 et une RMSE de 138.37.
- Le LSTM a obtenu une MAPE de 8.89%, une MAE de 31.79 et une RMSE de 36.93.
- Le 1D-CNN a montré des résultats avec une MAPE de 12.53%, une MAE de 43.12 et une RMSE de 52.45.
- Le méta-modèle a surpassé chacun des modèles individuels avec une MAPE de 8.08%, une MAE de 28.75 et une RMSE de 28.75.

Pour le deuxième ensemble de données (fournitures électroniques) :

- Le MLP a obtenu une MAPE de 15.84% , une MAE de 4.40 et une RMSE de 5.51.
- Le LSTM a obtenu une MAPE de 14.36%, une MAE de 4.03 et une RMSE de 33.36.
- Le 1D-CNN a montré des résultats avec une MAPE de 11.20%, une MAE de 3.80 et une RMSE de 4.78.
- Le méta-modèle a obtenu une MAPE de 13.50%, une MAE de 3.87 et une RMSE de 4.83.

- **Zeynep Hilal Kilimci,A. Okay Akyuz,Mitat Uysal,Selim Akyokus,M. Ozan Uysal,Berna Atak Bulbul et d Mehmet Ali Ekmiş(2019) [33]** ont abordé la prévision de la demande dans le domaine de la gestion de la chaîne logistique. L'objectif principal de cette étude est d'améliorer la précision des prévisions de la demande afin d'optimiser les stocks, réduire les coûts et augmenter les ventes ainsi que la fidélité des clients.

Ils ont proposé une approche intégrée combinant différentes méthodes de prévision, notamment les techniques d'analyse de séries temporelles, l'algorithme de régression par vecteur de support (SVR), XGBoost et des modèles d'apprentissage profond . Leur méthode inclut également une stratégie d'intégration décisionnelle inspirée du boosting, où les prévisions des modèles les plus performants sont combinées.

Leur approche a été évaluée sur un jeu de données provenant de SOK Market en Turquie, une

entreprise en pleine croissance avec 6700 magasins, 1500 produits et 23 centres de distribution. Le dataset comprenait 106 semaines de données de ventes pour 7888 produits distincts. Les résultats ont montré que l'approche proposée fournissait une MAPE moyenne de 24,7 %, démontrant une amélioration significative de la précision des prévisions par rapport aux méthodes traditionnelles. Ils ont aussi observé que l'ajout de modèles d'apprentissage profond dans leur système de prévision a augmenté la précision de 2 à 3,4 % pour certains groupes de produits.

- **Massimo Pacella et Gabriele Papadia**(2020) [11] ont examiné les défis posés par les dynamiques non linéaires et non stationnaires des séries temporelles dans la gestion de la chaîne logistique. L'objectif de leur étude est de proposer une technique visant à améliorer la prévision de la demande des clients finaux en utilisant des réseaux LSTM et BLSTM. Ces modèles ont été évalués sur un jeu de données couvrant une période temporelle incluant les années 2015, 2016 et 2017, et comprenait 10 séries temporelles distinctes, chacune représentant la demande mensuelle finale des clients pour un produit spécifique sur le marché. Les modèles proposés atteignent un RMSE variant entre environ 0,645 et 3,651 pour le LSTM, tandis que pour le BLSTM, le RMSE était compris entre environ 0,289 et 2,291.
- **Chien-Chih Wang, Chun-Hua Chien et Amy J. C. Trappey**(2021) [46] ont examiné l'incertitude de la demande chez les clients, leur objectif est d'améliorer la précision de la prédiction de la demande par les fournisseurs en utilisant les techniques ARIMA et LSTM. Ces modèles ont été évalués sur un jeu de données constitué des commandes de bacs IC (Integrated Circuits) pour les usines de production de semi-conducteurs, composé de statistiques quotidiennes sur le volume des ventes, totalisant 6000 données brutes. Ces données couvrent la période du 1er janvier 2017 au 31 août 2019 et concernent les cinq produits les plus vendus de l'entreprise étudiée. Pour les prédictions avec LSTM, les valeurs de RMSE varient entre 113,45 et 293,01, tandis que les valeurs de MAPE varient entre 0,2 et 28,3. En comparaison, pour les prédictions avec ARIMA, les valeurs de RMSE varient entre 1061,47 et 10273,37, tandis que les valeurs de MAPE varient entre 5 et 3015.
- **Oluwafunmi Adijat Elufioye, Chinedu Ugochukwu Ike, Olubusola Odeyemi, Favour Oluwadamilare Usman et Noluthando Zamanjomane Mhlongo** [26] (2024) ont étudié l'intégration des analyses prédictives basées sur l'intelligence artificielle dans les chaînes d'approvisionnement agricoles. Leur objectif était de renforcer la précision de la prévision de la demande et d'optimiser l'approvisionnement. Ils ont adopté une approche basée sur une revue systématique et une analyse de contenu pour évaluer l'impact des techniques d'IA, utilisant des algorithmes de machine learning et des analyses de big data. Cette approche intégrée vise à améliorer la prise de décision en rendant les chaînes d'approvisionnement plus efficaces, résilientes et durables. Les sources de données incluaient des bases de données académiques et des bibliothèques numériques telles que Google Scholar, IEEE Xplore et ScienceDirect. Bien que les jeux de données spécifiques utilisés ne soient pas détaillés, les technologies analysées montrent des améliorations notables dans

la gestion des chaînes d'approvisionnement agricoles.

- **Shuyun Ren, Hau-Ling Chan et Tana Siqin [38] (2019)** ont réalisé une revue de littérature approfondie sur les méthodes de prévision de la demande pour les produits de mode, en intégrant des méthodes statistiques avancées (ARIMA, SARIMA) et bayésiennes pour analyser les tendances de ventes historiques et prévoir la demande et des techniques d'IA notamment les réseaux de neurones artificiels (ANN) et les machines à apprentissage extrême (ELM), pour traiter de grands volumes de données et améliorer la précision des prévisions. Les résultats montrent que pour les produits de type basique, Company A, par exemple, utilise principalement des méthodes statistiques telles que l'auto-régression et ARIMA, ce qui permet des prévisions initiales basées sur des données quantitatives historiques, ajustées ensuite par des données qualitatives et l'expérience des experts pour obtenir des résultats finaux précis. Pour les produits de mode, où les données historiques sont insuffisantes, la prévision repose davantage sur l'analyse des tendances de mode et les informations générées par les clients, avec une forte dépendance à l'expertise humaine pour finaliser les résultats de prévision.
- **Javad Feizabadi (2020) [27]**, a exploré la prévision de la demande dans le domaine de la gestion de la chaîne logistique, visant à évaluer l'impact des méthodes de machine learning par rapport aux approches traditionnelles pour optimiser les performances logistiques. Feizabadi a introduit une méthode hybride combinant des données de séries temporelles avec des variables explicatives avancées utilisant des algorithmes de machine learning, notamment ARIMAX et les réseaux de neurones. Cette approche a été testée avec succès dans le secteur manufacturier de l'acier, montrant des gains significatifs en précision de prévision et en performance de la chaîne logistique comparative-ment aux méthodes traditionnelles. Les résultats des modèles ARIMAX (1,1,0) et ARIMAX (3,0,0) ont révélé une précision de prévision pour l'année 2017 variant entre 88,9% et 89,1%, soulignant ainsi l'efficacité de cette méthode dans des contextes industriels réels.
- **Ammar Mohamed Amer, Luh Putu Eka Yani, et I Made Alan Priyatna (2021) [16]**, ont exploré les applications de l'apprentissage automatique pour la prévision de la demande dans la gestion de la chaîne d'approvisionnement, visant à offrir une vue d'ensemble complète pour améliorer l'efficacité opérationnelle. Une revue systématique de la littérature a été effectuée sur les articles publiés entre 2010 et 2019. Sur les 1870 articles initialement identifiés, 77 ont été sélectionnés pour une analyse approfondie. Les algorithmes les plus couramment utilisés incluent les réseaux neuronaux (27 %), les réseaux neuronaux artificiels (22 %), les régressions vectorielles de support (18%) et les machines à vecteurs de support (10%). L'évaluation des résultats a révélé que les algorithmes de machine learning, en particulier les réseaux neuronaux et les régressions vectorielles de support, offrent des performances améliorées en termes de précision pour la prévision de la demande, surpassant les méthodes traditionnelles. Ils permettent une réduction significative des erreurs de prévision, illustrant une efficacité accrue dans la gestion des chaînes d'approvisionnement.

2.7 Tableau comparatif

Article	Approche proposée	Jeu de données	Résultats
[41]	Apprentissage en profondeur ensembliste (MLP, 1D-CNN, LSTM)	Séries temporelles couvrant les produits électroniques et sportifs de janvier 2015 à septembre 2017	dataset1 : MAE :28.75, RMSE : 28.75 dataset2 : MAE :3.87, RMSE : 4.83
[11]	LSTM et BLSTM	10 séries temporelles couvrant la demande mensuelle finale des clients pour des produits spécifiques de 2015 à 2017	RMSE LSTM : 0.645 - 3.651 RMSE BLSTM : 0.289 - 2.291
[46]	ARIMA et LSTM	Commandes de bacs IC pour 5 produits, 6000 données de ventes quotidiennes de 2017 à 2019	RMSE LSTM : 113.45 - 293.01 MAPE LSTM : 0.2 - 28.3 RMSE ARIMA : 1061.47 - 10273.37 MAPE ARIMA : 5 - 3015
[33]	Approche intégrée combinant l'analyse de séries temporelles, l'algorithme SVM et des modèles d'apprentissage profond.	Données provenant de SOK Market en Turquie, une entreprise avec 6700 magasins, 1500 produits et 23 centres de distribution.	MAPE moyenne de 24.7%
[27]	Une méthode hybride combinant : ARIMAX et les Réseaux de neurones	Données historiques d'une entreprise de fabrication d'acier de la demande sur trois ans (2018-2020)	ARIMAX (1,1,0) et ARIMAX (3,0,0). La précision de la prévision pour l'année 2017 varie entre 88,9% et 89,1% pour ARIMAX
[26]	Analyse prédictive pilotée par l'IA utilisant des méthodes de revue systématique et d'analyse de contenu	Non spécifié dans l'article	Les technologies utilisées rendent les chaînes d'approvisionnement agricoles en les rendant plus efficaces, résilientes et durables.

Article	Approche proposée	Jeu de données	Résultats
[38]	Méthodes statistiques avancées (ARIMA, SARIMA, Auto-régression, les techniques bayésiennes, les ANNs, les ELMs, Logique floue,	Non spécifié dans l'article.	Pour les produits de mode, où les données historiques sont insuffisantes, la prévision repose davantage sur l'analyse des tendances de mode et les informations générées par les clients, avec une forte dépendance à l'expertise humaine pour finaliser les résultats de prévision.
[16]	Les ANNs, SVR, SVM, ELM, Random Forest, Adaptive-neurofuzzy classifier, Back Propagation Network, Decision Tree, LR, LR, XGBoost	Non spécifié dans l'article.	Les algorithmes de ML, notamment les réseaux de neurones et les régressions vectorielles de support, sont souvent associés à des performances améliorées en termes de précision pour la prévision de la demande, par rapport aux méthodes traditionnelles.

TABLE 2.1 – Résumé des travaux connexes

2.8 Conclusion

Dans ce chapitre, nous avons présenté un état de l'art sur les modèles de prédiction de la demande et les techniques couramment utilisées en pratique. Nous avons également abordé un aperçu des séries chronologiques et des méthodes d'évaluation des modèles de prédiction, visant à mesurer leur précision et leur fiabilité dans la prédiction de valeurs futures. À la fin du chapitre, une revue de littérature a été présentée.

Proposition d'un modèle de prédiction de la demande dans une chaîne logistique

3.1 Introduction

Dans ce mémoire, nous proposons un modèle innovant pour la prédiction de la demande dans la chaîne logistique. Ce modèle combine les capacités de CNN et GRU grâce à un XGBoost. Ce chapitre est consacré à une présentation détaillée de la conception du modèle de prédiction proposé. Nous commencerons par présenter et visualiser le jeu de données, puis nous fournirons une explication détaillée du prétraitement effectué avant l'entraînement du modèle. Ensuite, nous décrirons l'architecture du modèle proposé et son développement. Enfin, nous présenterons les résultats obtenus et les comparerons à ceux de l'article [41].

3.2 Dataset

Le dataset utilisé pour cette étude comprend des données sur les demandes quotidiennes d'articles collectées sur une période de 5 ans. Il contient des informations sur 50 articles différents et couvre 10 magasins distincts. Il s'agit d'une série chronologique qui contient 913000 enregistrements et 4 colonnes qui sont date, store, item et demand. Chaque enregistrement représente une demande d'un article spécifique dans un magasin donné à une date précise entre le 1er janvier 2013 et le 31 décembre 2017.

Nous avons choisi ce dataset pour notre travail pour les raisons suivantes :

- **Pertinence :** Le dataset est pertinent pour notre recherche, il fournit une ressource précieuse pour l'utilisation grâce à sa nature multivariée, permettant une analyse détaillée des tendances de la demande à travers différents articles et magasins.

- **Taille** : La taille du dataset, avec ses 913 000 enregistrements, est suffisante pour garantir la robustesse des modèles, capturer une variété de comportements de demande, et améliorer la précision des prévisions.
- **Disponibilité** : le dataset est disponible publiquement sur Kaggle, une plateforme bien connue pour le partage de datasets et les compétitions de data science.

La figure 3.1 montre les dix premières lignes du jeu de données .

index	date	store	item	demand
0	2013-01-01 00:00:00	1	1	13
1	2013-01-02 00:00:00	1	1	11
2	2013-01-03 00:00:00	1	1	14
3	2013-01-04 00:00:00	1	1	13
4	2013-01-05 00:00:00	1	1	10
5	2013-01-06 00:00:00	1	1	12
6	2013-01-07 00:00:00	1	1	10
7	2013-01-08 00:00:00	1	1	9
8	2013-01-09 00:00:00	1	1	12
9	2013-01-10 00:00:00	1	1	9

FIGURE 3.1 – Les premières lignes du jeu de données

3.3 Visualisation de données

La visualisation des données est un processus essentiel pour créer des représentations visuelles des séries chronologiques. Ce processus permet de comprendre les distributions de données, d'obtenir des informations et de communiquer les résultats. Les figures 3.2, 3.3, 3.4 fournissent un aperçu des différentes facettes de la demande quotidienne, incluant la demande quotidienne globale, la demande par magasin et la demande par article.

La figure 3.2 présente les demandes quotidiennes agrégées par date. Chaque point sur le graphique représente la somme totale des demandes pour chaque jour, permettant une visualisation claire des variations de la demande au fil du temps. Cette visualisation permet de distinguer les tendances générales de la demande, ainsi que les variations saisonnières.

La figure 3.3 présente les demandes quotidiennes agrégées par magasin . Chaque point du graphique correspond à la demande quotidienne totale pour un magasin spécifique. Les variations observées dans les séries chronologiques des différents magasins permettent de visualiser les différences de comportement de la demande entre les emplacements.

La figure 3.4 présente les demande quotidienne agrégées par article. Chaque point du graphique représente la somme totale des demandes quotidiennes pour un article spécifique au fil du temps. Cette visualisation permet d'identifier les variations de la demande pour chaque article, capturant les tendances de consommation particulières à chaque produit.

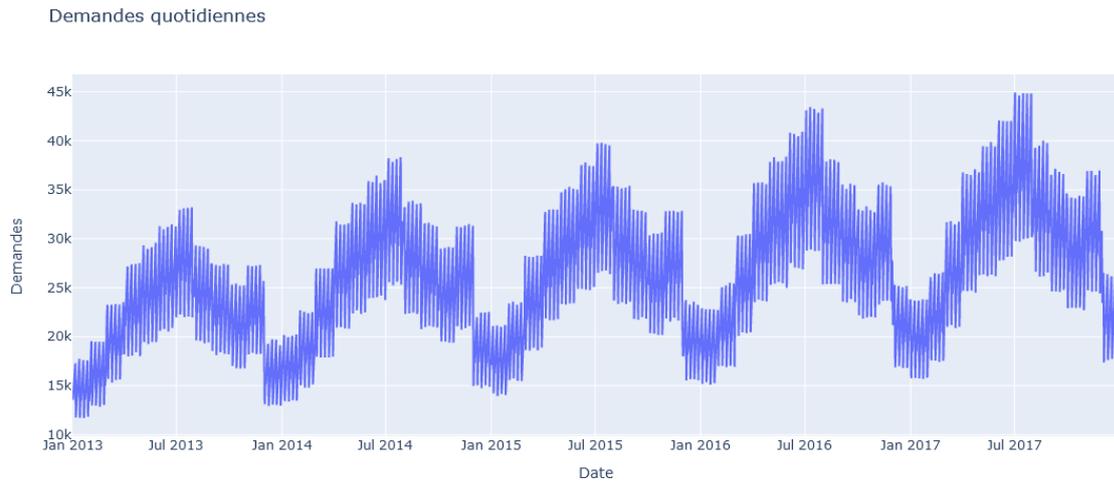


FIGURE 3.2 – La demande quotidienne

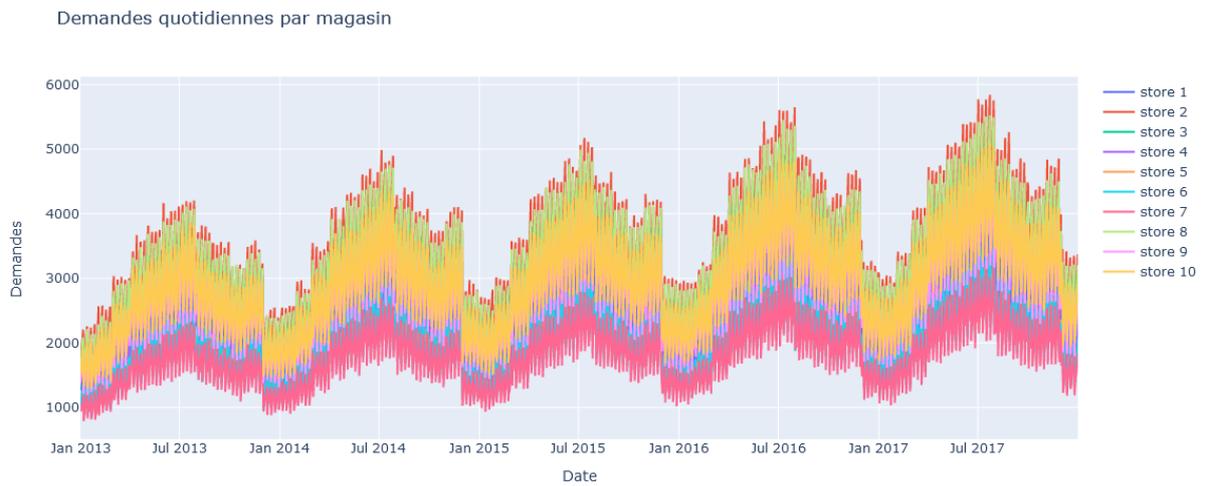


FIGURE 3.3 – La demande quotidienne par magasin

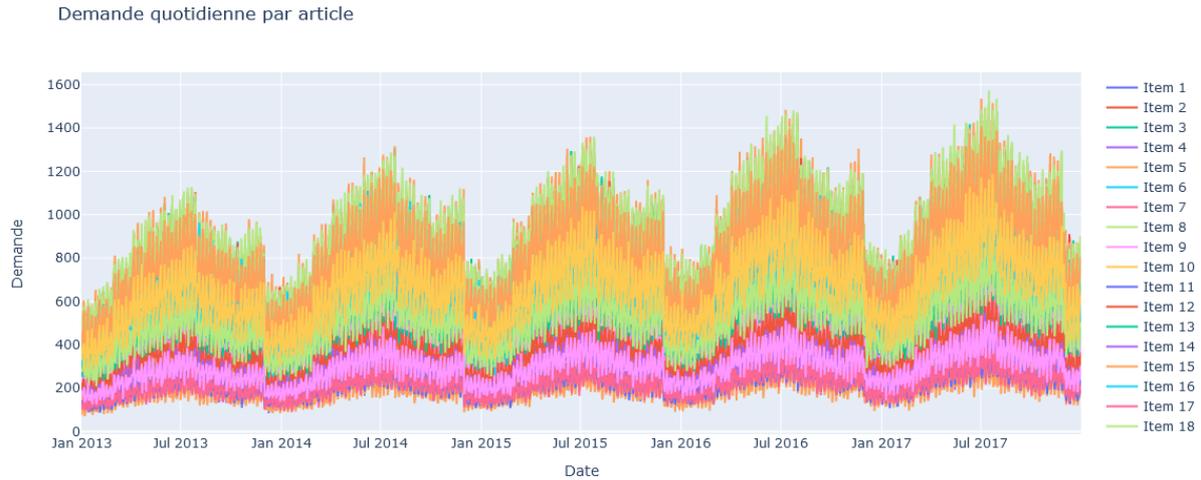


FIGURE 3.4 – La demande quotidienne par article

3.4 Préparation de données

Avant d'utiliser les modèles de prévision de la demande, nous avons réalisé plusieurs étapes de pré-traitement des données que nous allons détailler ci-dessous.

3.4.1 Nettoyage de données

Dans un premier temps, nous avons examiné le dataset afin de détecter la présence éventuelle de valeurs manquantes, nulles ou de doublons. Nous avons constaté qu'aucune valeur nulle ni aucun doublon n'étaient présents dans notre dataset, ce qui nous a permis de garantir la précision et la fiabilité de notre analyse en résolvant tout problème potentiel dès le début du processus.

3.4.2 Optimisation de l'analyse de la demande

Nous avons utilisé le regroupement des données par date, magasin et article comme une étape essentielle dans notre processus d'analyse de la demande. Cette technique consiste à rassembler les données brutes de demande quotidienne selon chaque combinaison unique d'article, de magasin et de date.

Pour chaque article demandé dans un magasin donné à une date particulière, nous calculons la demande moyenne quotidienne. L'objectif est de réduire le niveau de détail tout en préservant les informations essentielles pour comprendre les dynamiques du marché et identifier les tendances saisonnières.

3.4.3 Transformation des Données en Série Temporelle Supervisée

Nous avons transformé les données en un problème de série temporelle pour créer un ensemble de données supervisé adapté à l'entraînement de modèles de prédiction basés sur les réseaux de neurones profonds. Nous avons défini une fonction appelée `series_to_supervised` qui prend une série temporelle en entrée et la transforme en un ensemble de données supervisé .

En utilisant une fenêtre temporelle spécifiée, elle crée des séquences de données décalées dans le temps, comprenant les valeurs passées , la valeur actuelle , et les valeurs cibles décalées vers l'avant de lag pas de temps. Cette fonction permet d'organiser les données temporelles de manière structurée pour faciliter l'analyse des tendances passées et la prédiction des valeurs futures.

Dans notre étude, nous avons choisi une fenêtre de temps (`fenetre = 90`) et un décalage (`decalage = 90`), Cela signifie que nous avons structuré nos données temporelles en utilisant les 90 périodes précédentes pour prédire les valeurs exactement 90 périodes dans le futur.

Ensuite, nous avons supprimé les colonnes représentant les informations des magasins et des articles pour les périodes futures ($t+90$) pour éviter la fuite de données, où le modèle aurait accès à des informations futures qu'il ne devrait pas connaître. En éliminant ces colonnes, nous nous assurons que le modèle ne base pas ses prédictions sur des données qu'il ne devrait pas avoir.

Enfin, nous avons extrait la colonne contenant les valeurs futures de la demande, que nous souhaitons prédire, puis nous l'avons supprimée de l'ensemble de données supervisé pour l'utiliser comme étiquette de prédiction.

3.4.4 Fractionnement de Dataset

Dans notre étude, nous avons divisé l'ensemble de données supervisé en trois ensembles distincts illustré par la figure 3.5, nous avons d'abord séparé les données en un ensemble d'entraînement (80%) et un ensemble temporaire (20%). Ensuite, l'ensemble temporaire a été divisé en ensemble de validation (15%) et ensemble de test (5%).

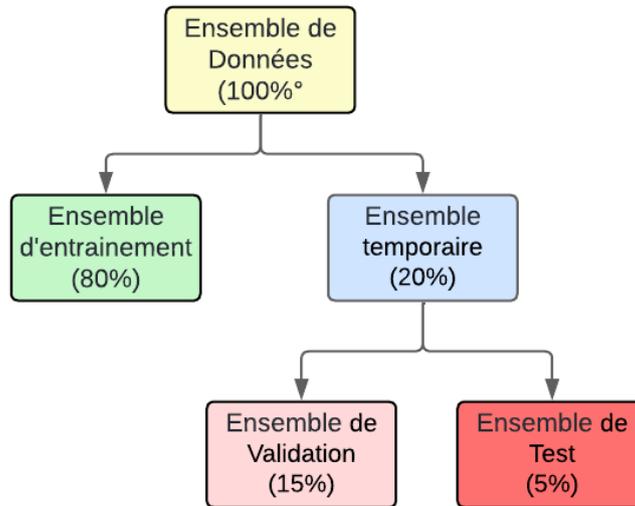


FIGURE 3.5 – Répartition des Données en Ensembles d'Entraînement, de Validation et de Test.

3.4.5 Normalisation de données

Pour garantir une échelle similaire entre différentes variables dans une série chronologique, c'est essentiel d'appliquer la normalisation des données. La méthode courante est la mise à l'échelle Min-Max, qui réduit les caractéristiques à un intervalle fixe $[0, 1]$.

La normalisation Min-Max est réalisée selon la formule suivante :

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

Dans la formule donnée, x représente la valeur de la caractéristique que nous cherchons à normaliser, tandis que x_{\min} et x_{\max} représentent respectivement la plus petite et la plus grande valeur observée pour cette caractéristique.

3.5 Modèle proposé

Nous avons développé une approche ensembliste illustrée par la figure 3.6 qui combine les prédictions de deux modèles neuronaux, GRU et CNN, à l'aide de XGBoost pour améliorer la précision de la prédiction. L'intégration de CNN et GRU via un métamodèle XGBoost permet d'obtenir une prédiction finale bénéficiant des points forts des deux types de modèles de base, améliorant ainsi la performance globale du système en termes de précision et de robustesse des prédictions. Cette approche est particulièrement avantageuse dans des contextes où les données présentent des caractéristiques à la fois locales et temporelles complexes, comme c'est souvent le cas dans la prédiction de la demande, grâce aux capacités

complémentaires de ces modèles.

Nous nous sommes inspirés des travaux de Mahya Seyedana et al [41], qui ont montré que la combinaison des prédictions de plusieurs modèles de base peut offrir des performances de généralisation comparables, voire supérieures, par rapport aux modèles individuels. Ces résultats soulignent l'importance d'adopter des méthodes de prévision en ensemble efficaces pour obtenir des solutions plus précises de prévision de la demande. Grâce à l'agrégation de prévisions diverses, l'apprentissage en ensemble peut atténuer les biais individuels des modèles et capturer une gamme plus large de motifs, améliorant ainsi la fiabilité et l'efficacité des stratégies de prévision de la demande.

- **GRU :**

Notre choix s'est porté sur le modèle GRU car il est capable de capturer les tendances et la saisonnalité présentes dans les données de séries temporelles. Sa capacité à modéliser les dépendances à long terme lui permet de saisir les variations sur des périodes prolongées. Cette capacité du GRU est particulièrement essentielle dans la prévision de la demande, où il est primordial d'anticiper avec précision les variations périodiques influencées par les saisons et les périodes de l'année.

- **1D-CNNs :**

Nous avons choisi les réseaux de neurones convolutifs (CNN) pour notre modèle de prédiction de la demande en raison de leur capacité à identifier et extraire des motifs temporels locaux. En appliquant des filtres le long de la dimension temporelle, les 1D-CNN peuvent capturer des caractéristiques pertinentes dans les données de séries temporelles. De plus, les fonctions d'activation utilisées dans les CNN introduisent une non-linéarité, permettant d'apprendre des représentations complexes et améliorant ainsi la précision des prédictions. Cette combinaison de détection de motifs temporels et de capacité à gérer la non-linéarité rend les 1D-CNN particulièrement adaptés pour les tâches de prévision de la demande [41].

- **Métamodèle XGBoost :**

XGBoost est utilisé comme métamodèle dans une approche ensembliste pour combiner les prédictions de deux modèles de base, CNN et GRU. Dans cette technique d'ensemble, les modèles de base sont entraînés de manière indépendante en parallèle, chacun produisant ses propres prédictions. Ensuite, les prédictions de CNN et GRU sont agrégées comme entrées pour XGBoost, qui génère la prédiction finale. Ce processus permet de capitaliser sur les forces respectives de chaque modèle de base afin d'améliorer la précision et la robustesse des prédictions.

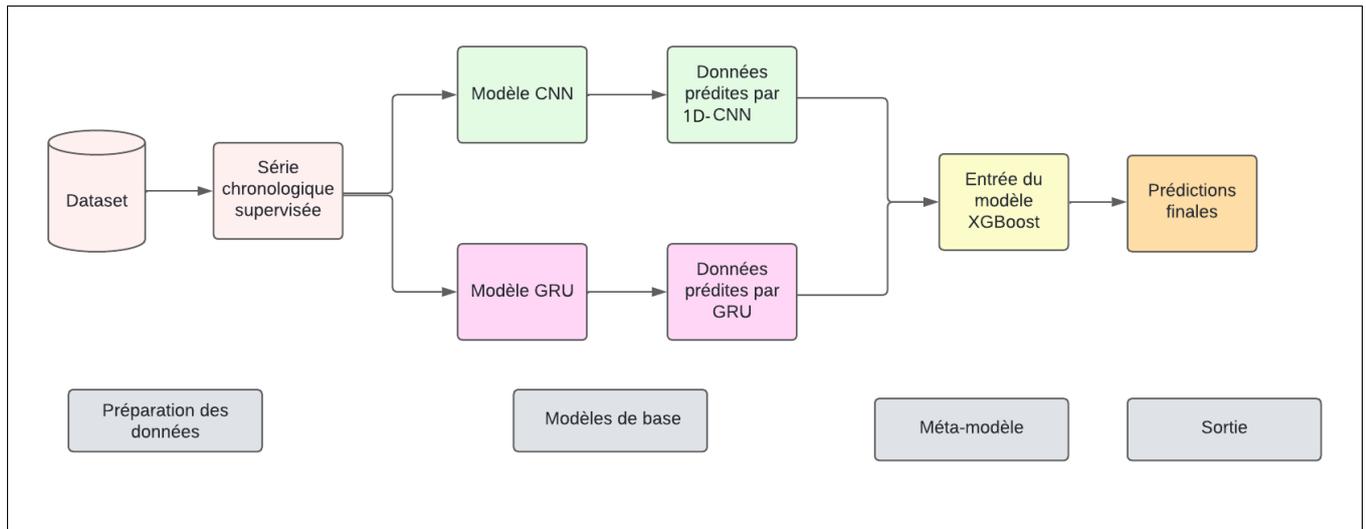


FIGURE 3.6 – Schéma du modèle ensembliste proposé avec CNN, GRU et XGBoost pour la prédiction de la demande.

3.6 Implémentation

Nous avons implémenté deux approches pour prédire la demande en utilisant le même jeu de données et en suivant un prétraitement uniforme. La première approche est basée sur celle décrite dans l'article [41], tandis que la seconde est notre propre modèle proposé. L'objectif de cette section est de comparer ces deux implémentations afin de déterminer laquelle est la plus efficace pour prédire la demande.

3.6.1 Environnement de développement

Nous avons utilisé l'environnement de développement intégré (IDE) Spyder pour notre projet. Spyder (Scientific Python Development Environment), qui fait partie de la distribution Anaconda, est un IDE open-source puissant conçu spécifiquement pour les scientifiques des données et les chercheurs. Il fournit une interface utilisateur riche avec des fonctionnalités avancées telles que l'édition de code, le débogage, la visualisation des données et l'intégration avec de nombreux packages Python, facilitant ainsi le développement, le test et l'optimisation des scripts Python. Nous avons choisi Spyder pour plusieurs raisons, son interface intuitive permet d'explorer les variables, d'éditer le code et de visualiser les graphiques de manière interactive ; son intégration avec IPython facilite l'exécution interactive du code Python avec des fonctionnalités comme l'auto-complétion et la gestion des sessions ; ses outils de débogage avancés, y compris un débogueur visuel pour inspecter les variables et corriger les erreurs ; enfin, sa compatibilité native avec les bibliothèques scientifiques telles que NumPy, SciPy, pandas, Matplotlib et scikit-learn, essentielles pour le développement, le test et l'optimisation de scripts Python dédiés à l'analyse de données.

3.6.2 Langage de programmation :

Le langage de programmation utilisé pour ce projet est Python, reconnu pour sa simplicité syntaxique et sa lisibilité, ce qui en fait un choix idéal pour les projets d'analyse de données et de machine learning. Python offre un écosystème de bibliothèques très riche, essentiel pour l'analyse de données. La communauté Python est active et offre un support substantiel à travers des forums, une documentation exhaustive et de nombreux tutoriels en ligne. En tant que langage interprété, Python permet une exécution interactive du code, ce qui est important pour le développement et le test itératif des modèles.

3.6.3 Bibliothèques utilisées :

Les bibliothèques que nous avons utilisées pour l'implémentation des modèles sont :

- **TensorFlow /Keras** : Nous avons utilisé cette bibliothèque qui permet de construire et d'entraîner efficacement des modèles de deep learning tels que les LSTM, MLP, CNN et GRU. Keras est l'API de haut niveau de TensorFlow dédiée à la création et à l'entraînement de modèles de deep learning. Elle est caractérisée par une interface conviviale et cohérente, optimisée pour les tâches courantes en deep learning. La modularité de Keras permet de composer des modèles en connectant des composants configurables avec flexibilité. De plus, elle facilite l'extension via la création de calques, de métriques et de fonctions de perte personnalisés, favorisant ainsi l'innovation et l'expérimentation dans la recherche en deep learning. Utilisée à la fois pour le prototypage rapide et pour des applications en production, Keras est un outil essentiel dans le domaine de l'apprentissage automatique moderne [19].

- **Scikit-learn** :

utilisé dans le prétraitement des données, la normalisation des données et l'évaluation des modèles. scikit-learn est un module Python dédié à l'apprentissage automatique, construit sur la base de SciPy et distribué sous la licence BSD 3-Clause. Initié en 2007 par David Cournapeau dans le cadre du programme Google Summer of Code, ce projet bénéficie depuis de nombreuses contributions volontaires. Actuellement maintenu par une équipe de bénévoles, scikit-learn nécessite Python (version ≥ 3.9) ainsi que d'autres dépendances telles que NumPy, SciPy, joblib et threadpoolctl pour son fonctionnement. Matplotlib est requis pour les fonctionnalités de visualisation, tandis que certaines fonctionnalités d'exemple peuvent nécessiter d'autres bibliothèques comme scikit-image, pandas, seaborn et plotly [40].

- **Pandas** : utilisé pour la manipulation et l'analyse des données nécessaires à la préparation des modèles. pandas est un package Python qui offre des structures de données rapides, flexibles et expressives conçues pour faciliter le travail avec des données "relationnelles" ou "étiquetées". Il se positionne comme le bloc de construction fondamental de haut niveau pour l'analyse de données

pratique et réelle en Python. En visant à devenir l'outil de manipulation et d'analyse de données open source le plus puissant et flexible disponible dans n'importe quel langage, pandas inclut des fonctionnalités telles que la gestion aisée des données manquantes, la mutabilité des tailles, l'alignement automatique et explicite des données, ainsi que des capacités avancées de groupement pour les opérations de split-apply-combine. Il propose également des outils robustes pour l'entrée/sortie de données à partir de fichiers plats, Excel et bases de données, ainsi que des fonctionnalités spécifiques aux séries temporelles comme la génération de plages de dates, la conversion de fréquence, les statistiques sur fenêtres mobiles et le décalage temporel [36].

- **NumPy** : utilisé pour le calcul numérique efficace et le traitement des tableaux multidimensionnels. NumPy est un projet open source animé par une communauté diversifiée de contributeurs. Son leadership est engagé à maintenir une communauté ouverte, inclusive et positive. NumPy encourage activement les contributions sous diverses formes, allant des petites améliorations aux corrections, jusqu'aux contributions plus substantielles au code source après consultation préalable via la liste de diffusion. En plus de l'écriture de code, il est possible de contribuer à NumPy en examinant les demandes de fusion, en gérant les problèmes nouveaux et anciens, en développant des tutoriels, des présentations et d'autres ressources pédagogiques, en améliorant le site web, en créant des designs graphiques pour les ressources de marque et matériaux promotionnels, en traduisant le contenu du site web, en participant à la sensibilisation et à l'intégration de nouveaux contributeurs, et en rédigeant des propositions de subventions et en aidant aux efforts de collecte de fonds [35].
- **XGBoost** : spécialisé dans les méthodes d'ensemble basées sur le gradient boosting. Cette bibliothèque de boosting de gradient distribué est largement utilisée pour entraîner des modèles d'apprentissage automatique. Elle combine les prédictions de multiples modèles faibles pour obtenir des résultats robustes, notamment dans la classification et la régression [29].
- **Matplotlib** : est une bibliothèque Python pour la visualisation de données. Elle est utilisée avec NumPy pour créer des graphiques dans des applications. Elle offre une interface de programmation orientée objet pour visualiser des données de manière graphique. Matplotlib a été développée par John D. Hunter et est maintenue par une communauté active [47].

3.6.4 Hyperparamètres

Pour l'entraînement de nos modèles, nous avons sélectionné les hyperparamètres suivants :

- **Époques** :

Une époque en apprentissage automatique (Machine Learning) fait référence à un passage complet de l'ensemble des données d'entraînement à travers l'algorithme. Un nombre d'époques élevé peut améliorer la précision du modèle, mais peut également entraîner un surapprentissage [23]. Dans

notre cas, nous avons choisi d'utiliser 50 époques pour trouver un équilibre entre l'optimisation de la précision et la prévention du surapprentissage.

◦ **batch_size :**

La taille de batch fait référence au nombre d'échantillons de données traités ensemble en une seule fois lors de l'entraînement en apprentissage automatique. Une taille de batch plus grande signifie que le modèle traite plus de données à la fois, ce qui peut accélérer l'entraînement mais pourrait nécessiter plus de mémoire. Dans notre choix, nous avons opté pour une taille de batch de 256 afin d'optimiser l'efficacité du processus d'entraînement tout en tenant compte de la gestion efficace des ressources disponibles.

◦ **Learning_rate :**

Le taux d'apprentissage est un hyperparamètre essentiel en apprentissage automatique, influençant la vitesse à laquelle la descente de gradient converge vers un minimum local. Il représente le coefficient par lequel les poids du modèle sont ajustés à chaque itération de l'entraînement. Un taux d'apprentissage trop élevé peut entraîner une convergence rapide mais risque de sauter le minimum global, tandis qu'un taux trop bas peut ralentir la convergence [10]. Dans notre cas, nous avons utilisé un taux d'apprentissage de 0.0003, choisi après des essais pour optimiser les performances du modèle.

◦ **optimiseur Adam :**

L'optimiseur Adam est un algorithme d'optimisation itératif largement utilisé pour minimiser la fonction de perte lors de l'entraînement des réseaux neuronaux. Créé par Diederik P. Kingma et Jimmy Ba en 2014, Adam combine les caractéristiques de RMSprop et de la descente de gradient stochastique avec momentum. Il ajuste le taux d'apprentissage en fonction des gradients au carré tout en utilisant une moyenne mobile du gradient pour bénéficier du momentum. Cette approche dynamique et adaptative permet à Adam d'optimiser efficacement les paramètres du réseau, favorisant ainsi la convergence vers le minimum global [45].

◦ **Fonction d'activation RELU :**

La fonction d'activation ReLU (Rectified Linear Unit) est définie par $\text{ReLU}(x) = \max(0, x)$, transformant les valeurs négatives en zéro et laissant les valeurs positives inchangées. Elle ne sature pas pour les valeurs positives, ce qui permet de mieux capturer les relations non linéaires dans les données et favorise un apprentissage plus rapide et efficace. ReLU est également plus rapide à calculer que les fonctions sigmoïdes et tanh, réduisant ainsi le temps de calcul global et permettant de former des modèles profonds plus efficacement [4].

- **Dropout** : Dropout est une technique de régularisation utilisée lors de l'entraînement des réseaux de neurones, où à chaque étape d'entraînement, chaque neurone (sauf les neurones de sortie) a une probabilité p d'être temporairement « éteint ». Cela signifie que le neurone est complètement ignoré pendant cette étape d'entraînement spécifique, mais il peut être actif à nouveau lors des étapes suivantes. Le taux d'extinction p (dropout rate) contrôle la proportion de neurones désactivés à chaque étape, contribuant ainsi à prévenir le surapprentissage. Nous avons spécifiquement utilisé Dropout dans les modèles LSTM et GRU pour régulariser la complexité des réseaux récurrents. Ces derniers sont particulièrement sensibles aux données séquentielles, nécessitant une régularisation efficace afin d'améliorer la généralisation et de maintenir des performances stables lors de l'entraînement sur des séquences temporelles [2].
- **Recherche en grille** :

La recherche en grille est une méthode systématique pour sélectionner les hyperparamètres d'un modèle machine learning en évaluant toutes les combinaisons possibles des valeurs spécifiées pour chaque hyperparamètre. Chaque combinaison est évaluée par l'entraînement de modèles distincts, et celle qui offre les meilleurs résultats sur les données de validation est choisie. Nous avons utilisé cette approche avec XGBoost pour identifier les hyperparamètres optimaux de notre modèle, en explorant toutes les combinaisons possibles et en sélectionnant celles qui présentent les meilleures performances sur les données de validation [2].

3.6.5 Entraînement des modèles de base

Nous avons entraîné les modèles de base utilisés dans l'article [41], à savoir LSTM, CNN et MLP, ainsi que les modèles de base de notre approche proposée, qui sont GRU et CNN

- **LSTM** :

Le modèle LSTM se compose de deux couches LSTM suivies d'une couche Dense pour la sortie, et de deux couches de Dropout placées après chaque couche LSTM.

La première couche LSTM, avec 64 neurones et une activation ReLU, prend en entrée des séquences de données temporelles et retourne des vecteurs. Cette couche permet de capturer les dépendances temporelles dans les données.

Ensuite, une deuxième couche LSTM avec 32 neurones, également avec une fonction d'activation ReLU, reçoit ces vecteurs et les résume en un vecteur de taille 32, fournissant ainsi des représentations plus compactes et approfondies de l'information temporelle.

La couche de sortie est une couche Dense qui transforme ce vecteur en une seule valeur prédictive.

Pour régulariser le modèle et éviter le surapprentissage, une couche Dropout est placée après chaque

couche LSTM.

○ **CNN :**

Le modèle CNN se compose d'une couche Conv1D, une couche MaxPooling1D, une couche Flatten, et enfin de deux couches Dense.

Tout d'abord la couche de convolution à une dimension avec 64 filtres, une fenêtre de taille 2 et une fonction d'activation RELU, prend en entrée des séquences de données temporelles et produit un tenseur qui représente les caractéristiques extraites par les filtres de convolution appliqués à la séquence d'entrée. Cette couche permet de capturer les motifs locaux dans la série temporelle.

Ensuite une couche MaxPooling intervient pour réduire la dimensionnalité de chaque canal de sortie de Conv1D en prenant la valeur maximale sur des fenêtres de taille 2 afin de conserver les caractéristiques les plus pertinentes tout en réduisant la complexité des données.

La couche suivante est une couche flatten, elle permet de transformer la sortie de MaxPooling1D en un vecteur plat.

Les deux dernières couches sont des couches Dense. La première couche dense possède 50 neurones avec une activation ReLU, qui joue un rôle dans la combinaison des caractéristiques extraites par les couches précédentes, et la deuxième couche dense génère la prédiction finale.

○ **MLP :**

Le modèle MLP comporte deux couches Dense, la première avec 100 neurones et une activation RELU, qui prend en entrée une séquence de données temporelles. Son rôle est d'extraire des caractéristiques significatives des séquences temporelles, permettant au modèle d'apprendre des représentations complexes et non linéaires des données.

Ensuite, la sortie de cette première couche est transmise à la deuxième couche Dense, qui agit comme la couche de sortie du modèle. Cette couche génère des prédictions finales pour chaque exemple d'entraînement.

○ **GRU :**

Le modèle GRU se compose de deux couches GRU suivies d'une couche Dense pour la sortie, et de deux couches de Dropout placées après chaque couche GRU.

La première couche GRU, avec 64 unités et une activation ReLU, prend en entrée des séquences de données temporelles et retourne des séquences de vecteurs. Cette couche est efficace pour capturer les dépendances temporelles dans les données.

Ensuite, une deuxième couche GRU avec 32 unités, également avec une activation ReLU, reçoit les

séquences de vecteurs de la première couche GRU. Elle résume ces séquences en un vecteur de taille 32, fournissant ainsi des représentations plus compactes et approfondies de l'information temporelle.

La couche de sortie est une couche Dense qui transforme ce vecteur en une seule valeur prédictive.

Pour régulariser le modèle et prévenir le surapprentissage, une couche de Dropout avec un taux de 20% est placée après chaque couche GRU. Cela aide à réduire la sensibilité du modèle aux détails spécifiques des échantillons et améliore sa capacité à généraliser sur de nouvelles données.

3.6.6 Entraînement des modèles ensemblistes :

- **Le Méta-modèle proposé dans l'article [41] :**

Pour implémenter le méta-modèle proposé dans l'article, nous avons combiné les prédictions de chaque modèle LSTM, CNN, MLP pour créer une méta-caractéristique, que nous avons ensuite divisée en ensembles d'entraînement et de validation. Le méta-modèle est conçu sur le modèle MLP avec deux couches Dense : la première possède 100 neurones avec une activation ReLU, suivie d'une couche de sortie Dense pour générer la prédiction finale. le modèle final a été évalué sur les mêmes données de test pour mesurer son efficacité à prédire les valeurs cibles.

- **le Meta-modèle proposé :**

Pour améliorer la précision de nos prédictions, nous avons développé un méta-modèle en combinant les prédictions des modèles CNN et GRU. Ce méta-modèle XGBoost utilise les prédictions des modèles individuels CNN et GRU comme nouvelles caractéristiques d'entrée. Nous avons utilisé la recherche par grille pour sélectionner les meilleurs hyperparamètres du modèle XGBoost, afin d'optimiser ses performances. Après entraînement sur un ensemble d'entraînement et de validation dérivé des prédictions des modèles individuels, le méta-modèle a été évalué sur les données de test pour mesurer son efficacité à prédire les valeurs cibles.

3.7 Evaluation des modèles

Pour évaluer la performance des modèles et mesurer précisément la qualité de leurs prévisions, nous avons choisi d'utiliser le RMSE, le MAE et le coefficient de détermination R^2 . Chaque métrique offre une perspective unique sur la précision et l'adaptabilité du modèle aux données.

Le RMSE est particulièrement utile lorsque les grandes erreurs ont un impact plus important sur la performance globale du modèle, le MAE est utile lorsque toutes les erreurs, qu'elles soient grandes ou petites, contribuent de manière égale à l'évaluation de la performance. Enfin, le R^2 mesure dans quelle mesure le modèle s'adapte bien au problème de prédiction.

Afin de vérifier si les modèles entraînés ne souffrent ni de surapprentissage, qui se produit lorsque le modèle généralise trop et mémorise le bruit des données d'entraînement, ni de sous-apprentissage, où le modèle n'apprend pas suffisamment et ne peut pas bien généraliser, nous avons tracé les courbes de pertes sur les données d'entraînement et de validation pour chaque modèle, voir les figures 3.7, 3.8, 3.9 et 3.10. Nous remarquons que les courbes suivent une tendance similaire. La perte d'entraînement diminue régulièrement au fur et à mesure des époques, ce qui indique que le modèle apprend bien à partir des données d'entraînement, La perte de validation diminue également, et cette diminution se produit de manière cohérente avec la perte d'entraînement. nous remarquons qu'il n'y a pas d'écart significatif entre les deux courbes ce qui signifie que le modèle généralise bien sur les données de validation et ne mémorise pas simplement les données d'entraînement.

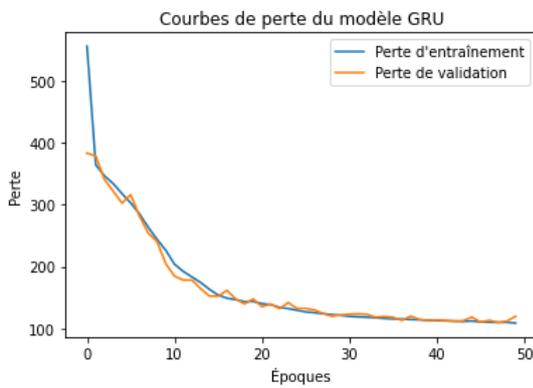


FIGURE 3.7 – Courbes de perte du Modèle GRU

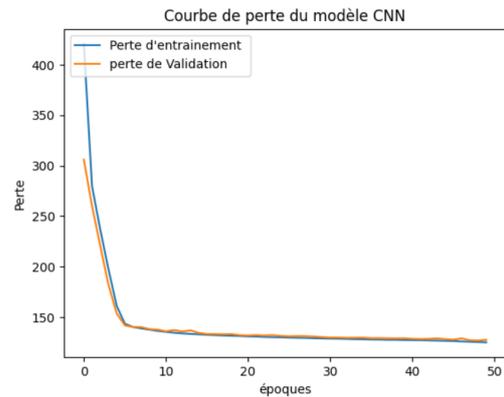


FIGURE 3.8 – Courbes de perte du Modèle CNN

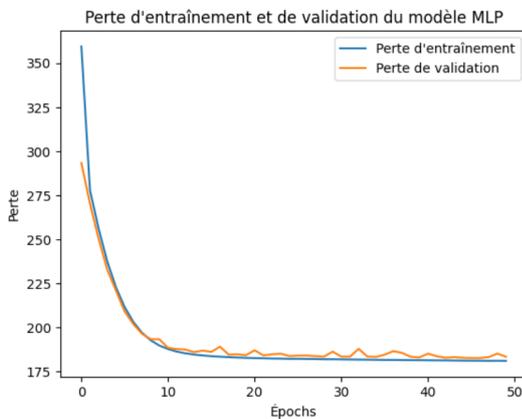


FIGURE 3.9 – Courbes de perte du Modèle MLP

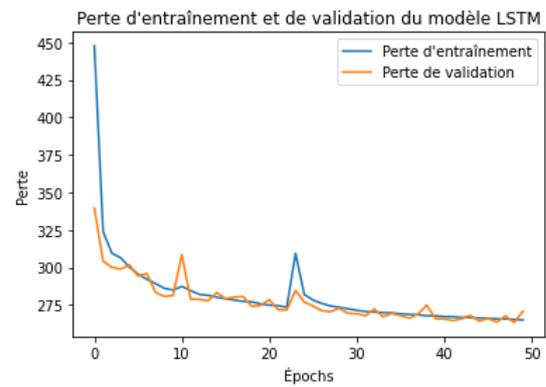


FIGURE 3.10 – Courbes de perte du Modèle LSTM

3.8 Résultats et discussion

Dans cette section, nous présentons les résultats de l'évaluation des modèles entraînés. Nous commençons par les résultats des modèles individuels : GRU, LSTM, CNN, MLP. Ensuite, nous exposons les résultats des approches ensemblistes utilisées dans l'article [41] ainsi que ceux de XGBoost, que nous avons également présenté. Enfin, nous procéderons à une comparaison entre ces deux approches.

3.8.1 Résultats des modèles individuels

Nous résumons les résultats des métriques RMSE, MAE et R2 pour chaque modèle (CNN, LSTM, GRU et MLP) dans le tableau 3.1 ci-dessous .

Modèle	RMSE	MAE	R ²
GRU	9.99	7.35	0.88
LSTM	13.22	9.91	0.78
CNN	11.61	8.64	0.83
MLP	11.44	8.47	0.84

TABLE 3.1 – Performance des modèles de prédiction avec les métriques RMSE, MAE et R².

3.8.2 Résultats des approches ensemblistes

Ci-dessous se trouve un résumé des résultats de notre approche proposée et de celle de l'article, évalués selon les métriques RMSE, MAE et R2, présentés dans le tableau 3.2 . Les performances pour notre approche et celle de l'article sont respectivement les suivantes : RMSE (9,70 - 11,30), MAE (8,40 - 7,19) et R2 (0,83 et 0,89).

Modèle	RMSE	MAE	R ²
Approche de l'article	11.30	8.40	0.84
Notre Approche	9.70	7.19	0.89

TABLE 3.2 – Performance des Approches ensemblistes

3.8.3 Comparaison des résultats entre l'approche proposée et celle de l'article [41]

Les résultats des évaluations de performance montrent une différence claire entre l'approche décrite dans l'article et notre propre méthode. En termes de précision de prédiction, notre approche se distingue avec un RMSE de 9.70 et un MAE de 7.19, comparé à 11.30 et 8.40 respectivement pour l'approche de l'article illustrés par les figures 3.11 et 3.12 . Ces chiffres indiquent que notre modèle produit des prédictions avec une erreur moyenne absolue et quadratique inférieure, suggérant une meilleure capacité à estimer les valeurs cibles. De plus, notre modèle démontre une meilleure adéquation aux données avec un R2 de 0.89 contre 0.84 pour l'approche de l'article voir la figure 3.13, ce qui témoigne d'une meilleure capacité à expliquer la variance des données observées. Ces résultats cumulés suggèrent que notre approche pourrait être plus adaptée pour ce problème spécifique, offrant potentiellement des prédictions plus précises et une meilleure performance globale par rapport à l'approche existante dans l'article [41].

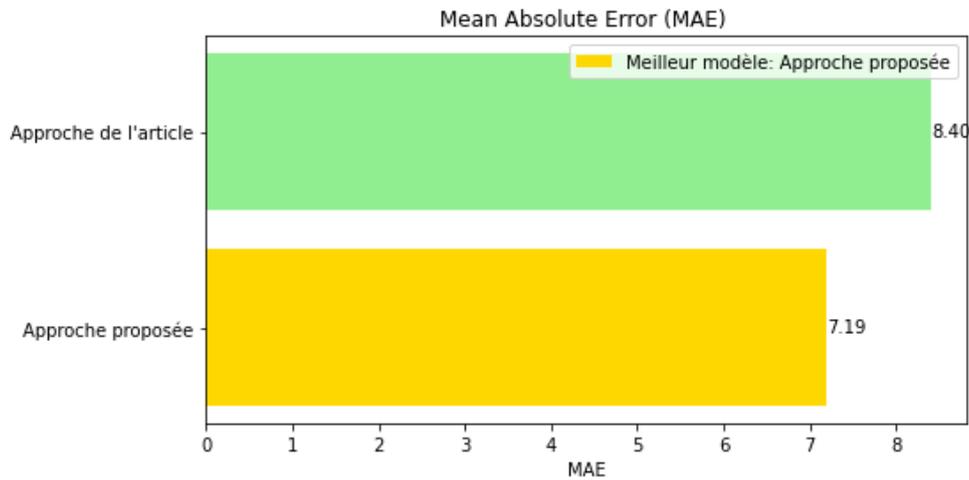


FIGURE 3.11 – Erreur quadratique moyenne

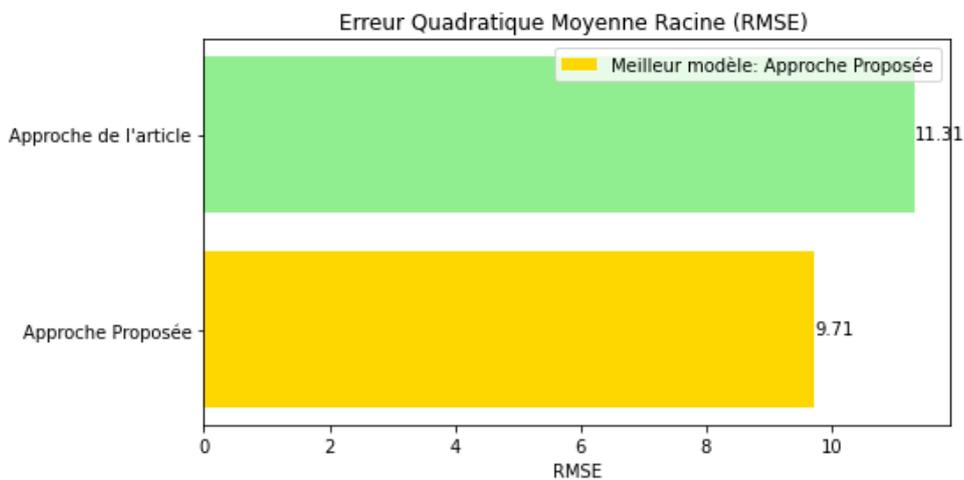
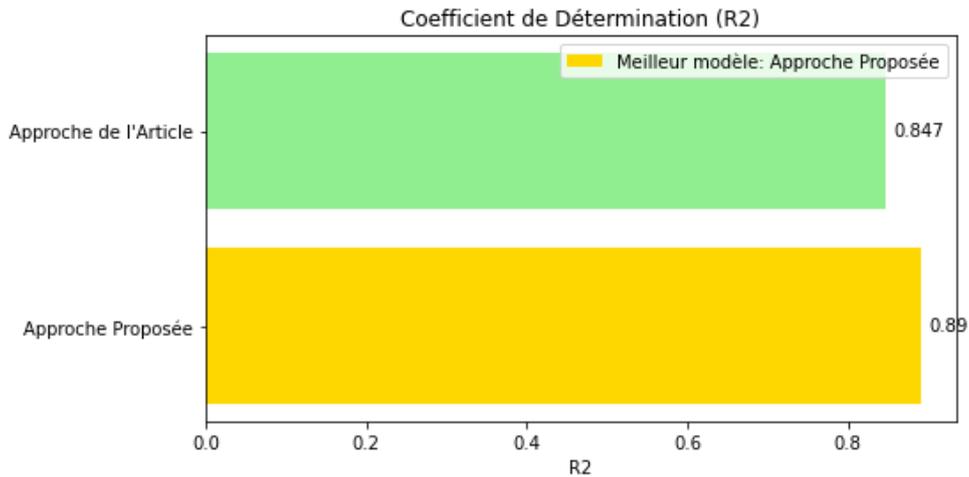


FIGURE 3.12 – Racine de l'erreur quadratique moyenne


 FIGURE 3.13 – Coefficient de Détermination R^2 (R-squared)

Notre choix de comparer ces métriques spécifiques (RMSE, MAE, R2) et notre interprétation des résultats reposent sur plusieurs considérations méthodologiques et sur les caractéristiques uniques de notre méthode par rapport à celle de l'article. Notamment, notre approche intègre ... Dans notre approche, nous avons entraîné plusieurs modèles de base ainsi qu'un méta-modèle pour comparer leur performance à ceux décrits dans l'article [41]. Voici un résumé des différences clés entre les deux approches, celle de l'article et la notre :

Modèles de base :

- **LSTM, 1D-CNN, MLP (de l'article [41])** : Nous avons reproduit l'entraînement des modèles LSTM, CNN et MLP tels qu'ils sont décrits dans l'article. Chaque modèle a été configuré selon les spécifications détaillées dans la littérature existante, avec des ajustements mineurs pour l'adaptation à notre ensemble de données spécifique.
- **GRU et 1D-CNN (notre approche)** : En plus des modèles traditionnels comme LSTM et MLP, nous avons introduit l'utilisation de GRU, une variante de réseau récurrent capable de capturer efficacement les dépendances temporelles dans les données. Notre choix d'intégrer GRU repose sur sa performance empirique supérieure dans d'autres contextes similaires. Nous avons également employé 1D-CNN, reconnu pour son efficacité dans l'extraction de caractéristiques locales dans les séquences temporelles, complétant ainsi notre gamme de modèles de base.

Entraînement des modèles ensemblistes :

- **Méta-modèle de l'article [41]** : Nous avons suivi la méthode proposée dans l'article pour construire un méta-modèle basé sur les prédictions combinées des modèles LSTM, CNN et MLP. Ce méta-modèle utilise un modèle MLP pour fusionner ces prédictions et générer une prédiction finale.
- **Notre méta-modèle** : Pour améliorer encore la précision des prédictions, nous avons développé un

méta-modèle en utilisant les prédictions des modèles CNN et GRU comme caractéristiques d'entrée pour un modèle XGBoost. Cette approche vise à exploiter les forces complémentaires des modèles CNN et GRU dans la prédiction des valeurs cibles.

Justification de la comparaison :

Les différences observées dans les performances (mesurées par RMSE, MAE et R2) entre notre approche et celle de l'article reflètent non seulement les variations dans les modèles de base utilisés, mais aussi les stratégies distinctes pour combiner les prédictions dans nos méta-modèles respectifs. Notre décision de comparer ces métriques spécifiques était motivée par leur pertinence dans l'évaluation de la précision des prédictions et de l'adaptation aux données observées.

3.8.4 Valeurs réelles et valeurs prédites

Pour mieux visualiser la performance de notre modèle, la figure 3.14 ci-dessous compare les valeurs réelles aux valeurs prédites par notre approche.

Il est clairement observable que notre approche démontre une capacité remarquable à effectuer des prédictions précises, suivant de près les points de données réels.

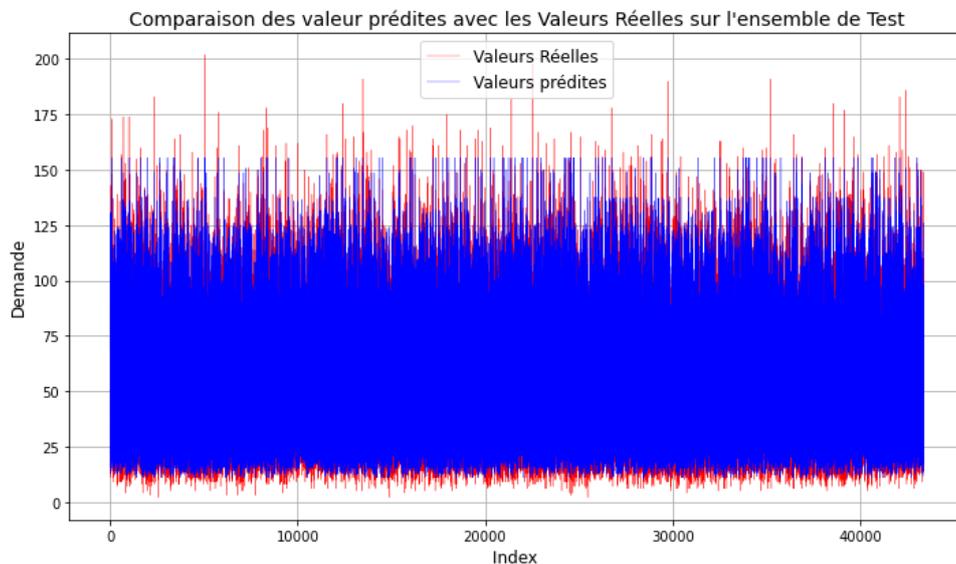


FIGURE 3.14 – Résultats de prédiction sur les données de test

3.8.5 Schéma récapitulatif de notre approche proposée

Pour récapituler notre méthode proposée, nous illustrerons les étapes clés dans un schéma détaillé, présenté à la figure, comme discuté précédemment dans ce chapitre. 3.15

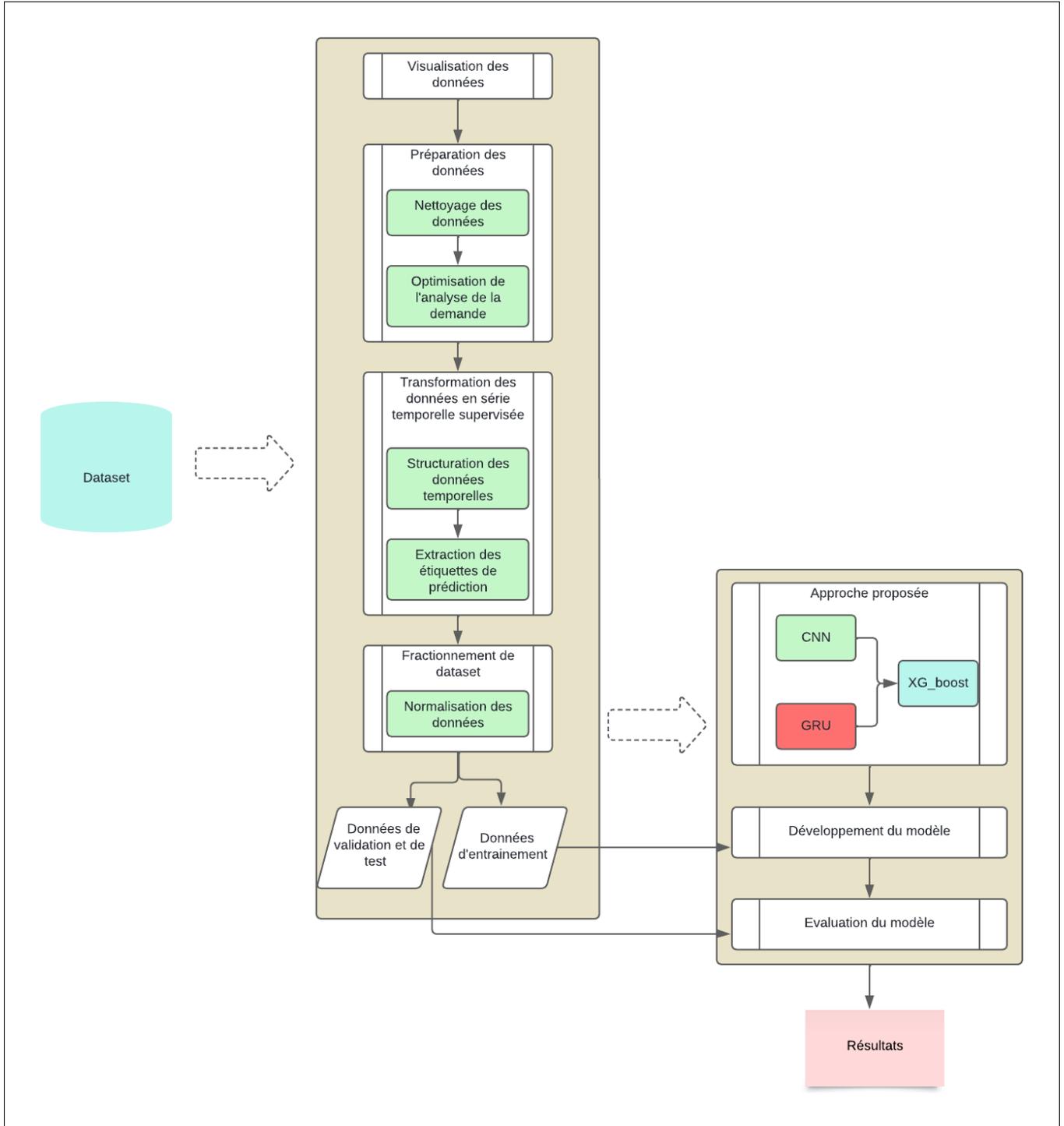


FIGURE 3.15 – Méthodologie proposée de travail

3.9 Conclusion

Dans ce chapitre, nous avons abordé les principaux aspects de notre étude portant sur l'utilisation des modèles d'apprentissage automatique pour la prédiction de la demande. Au cours de cette étude, nous avons exploré les différentes étapes nécessaires à une analyse approfondie. Nous avons détaillé la méthodologie suivie, depuis la visualisation des données jusqu'à l'évaluation des modèles et l'obtention des résultats, en incluant une comparaison avec l'article [41] .

Conclusion générale & perspectives

Dans ce mémoire, nous avons proposé un modèle de prédiction de la demande dans une chaîne logistique, en intégrant des techniques d'intelligence artificielle et d'apprentissage automatique, spécifiquement des réseaux de neurones et l'analyse des séries temporelles. Notre objectif était de répondre aux défis identifiés dans la littérature, notamment ceux liés à la qualité des données et à la sélection des modèles appropriés.

Pour garantir la qualité des données, nous avons effectué une analyse approfondie incluant le nettoyage et la normalisation des données. Notre approche a combiné les sorties de deux types de réseaux de neurones, le 1D CNN et le GRU, intégrées dans un modèle XGBoost pour prédire la demande logistique. Les CNN 1D identifient les motifs locaux dans les données temporelles, tandis que les GRU modélisent les dépendances à long terme de manière plus simple et rapide que les LSTM. XGBoost gère les données complexes et améliore la précision des prédictions en intégrant les résultats des CNN 1D et des GRU. Les résultats expérimentaux démontrent l'efficacité de cette approche d'ensemble, montrant des améliorations significatives dans la précision des prévisions de la demande en exploitant les forces de chaque modèle. En particulier, les mesures d'évaluation telles que la Racine de l'Erreur Quadratique Moyenne (RMSE), l'Erreur Absolue Moyenne (MAE) et le Coefficient de Détermination (R^2) reflètent les performances élevées et la fiabilité de notre modèle prédictif. En surmontant les défis liés à la qualité des données et à la sélection des modèles, notre contribution au domaine de la prédiction de la demande logistique est notable. Notre modèle ensembliste, combinant CNN 1D, GRU et XGBoost, a montré une efficacité supérieure avec un RMSE de 9,70, un MAE de 7,19 et un R^2 de 0,89, surpassant les performances de l'approche de l'article comparatif. Ces résultats ont des implications significatives pour la gestion des stocks et l'optimisation des chaînes logistiques, facilitant une prise de décision améliorée.

Pour les travaux futurs, il serait bénéfique d'explorer davantage l'impact des conditions économiques et des facteurs environnementaux sur la demande prédite. L'intégration de données macroéconomiques telles que le PIB "Produit Intérieur Brut" qui est une mesure économique clé qui évalue la valeur totale des biens et services produits dans un pays sur une période donnée, souvent une année, et est utilisé

comme indicateur de la santé économique, les indices de confiance des consommateurs et les tendances climatiques pourrait enrichir notre modèle prédictif. En analysant ces variables externes, nous pourrions mieux anticiper les fluctuations saisonnières et les changements structurels affectant la demande, offrant ainsi une gestion des stocks encore plus robuste et adaptative. De plus, l'évaluation continue de la qualité des données et l'adoption de techniques avancées de prétraitement pourraient améliorer la fiabilité et la précision de nos prévisions, renforçant ainsi l'applicabilité et l'efficacité opérationnelle de notre approche dans différents contextes logistiques et géographiques.

Bibliographie

- [1] 10.2. gated recurrent units (GRU) — dive into deep learning 1.0.3 documentation.
- [2] Cours : M2 IA (apprentissage avancé) / dr bouchebbah.
- [3] Ensemble learning in machine learning : Stacking, bagging and boosting. Image and description retrieved from the source. Consulté le 25/06/2024.
- [4] Fonction d'activation : Comment elle fonctionne? - meilleur guide. Section : Deep Learning.
- [5] Introduction to ANN | set 4 (network architectures). Section : Computer Subject.
- [6] LA CHAINE LOGISTIQUE ET LA GESTION DES STOCKS d'UNE ENTREPRISE.pdf.
- [7] Poly_sc.pdf.
- [8] Time-series forecasting with deep learning : a survey.
- [9] Wang, j., swartz, c. l. e., & huang, k. (2023). deep learning-based model predictive control for real-time supply chain optimization. journal of process control, 129, 103049. doi : 10.1016/j.jprocont.2023.103049 - recherche google.
- [10] What is learning rate in machine learning? the full guide.
- [11] Evaluation of deep learning with long short-term memory networks for time series forecasting in supply chain management - ScienceDirect, 2020.
- [12] Simple 1D convolutional neural network (CNN) architecture with two convolutional layers, 2020. Consulté le 2024-06-25.
- [13] Supply chain model, 2021. Consulté le 06 juillet 2024.
- [14] Logística e supply chain, 2022. Consulté le 06 juillet 2024.
- [15] . Modèle d'ensemble. Image, 2024. Capture d'écran extraite de <https://www.ia-insights.fr/ensembling-en-ia-definition-et-fonctionnement/>.
- [16] Ammar Mohamed Aamer, Luh Putu Eka Yani, and I Made Alan Priyatna. Data analytics in the supply chain management : Review of machine learning applications in demand forecasting. *Operations and Supply Chain Management*, 14(1) :1–13, 2021.
- [17] Anonyme. Random forest algorithm in machine learning. <https://www.geeksforgeeks.org/random-forest-algorithm-in-machine-learning/>. Consulté le 12 juin 2024.

-
- [18] Anonymous. Time series analysis : Arma, arima, sarima, 2024.
- [19] TensorFlow Authors. Guide de tensorflow keras, 2020.
- [20] Martin Beaulieu and Jacques Roy. Optimisation de la chaîne logistique et productivité des entreprises. https://cpp.hec.ca/cms/assets/documents/recherches_en_cours/optimisation-de-la-chaîne-logistique-et-productivite-des-entreprises.pdf, année de publication. Consulté le 18 juin 2024.
- [21] Ismahane BELKACEM Bouzida. Spécialité : Logistique et commerce international.
- [22] Ottavio Calzone. An intuitive explanation of LSTM.
- [23] Daniel. Epoch : An essential notion in real-time programming.
- [24] Utpal Kumar Das, Kok Soon Tey, Mehdi Seyedmahmoudian, Saad Mekhilef, Moh Yamani Idna Idris, Willem Van Deventer, Bend Horan, and Alex Stojcevski. Forecasting of photovoltaic power generation and model optimization : A review. *Renewable and Sustainable Energy Reviews*, 81 :912–928, 2018.
- [25] Chuning Deng and Yongji Liu. A deep learning-based inventory management and demand prediction optimization method for anomaly detection. *Wireless Communications and Mobile Computing*, 2021 :Article ID 9969357, 2021.
- [26] Oluwafunmi Adijat Elufioye, Chinedu Ugochukwu Ike, Olubusola Odeyemi, Favour Oluwadamilare Usman, and Noluthando Zamanjomane Mhlongo. Ai-driven predictive analytics in agricultural supply chains : A review : Assessing the benefits and challenges of ai in forecasting demand and optimizing supply in agriculture. 5(2) :473–497, 2024. Publisher : Fair East Publishers.
- [27] Javad Feizabadi. Machine learning demand forecasting and supply chain performance. 25(2) :119–142, 2020.
- [28] Julien Francois. Planification des chaînes logistiques : modélisation du système décisionnel et performance.
- [29] GeeksforGeeks. XGBoost. <https://www.geeksforgeeks.org/xgboost/>. Consulté le 12 juin 2024.
- [30] Data Headhunters. Xgboost vs lightgbm : Gradient boosting in the spotlight, 2024. Publié le 05 Janvier 2024.
- [31] Olorunyomi Stephen Joel, Adedoyin Tolulope Oyewole, Olusegun Gbenga Odunaiya, and Oluwatobi Timothy Soyombo. LEVERAGING ARTIFICIAL INTELLIGENCE FOR ENHANCED SUPPLY CHAIN OPTIMIZATION : A COMPREHENSIVE REVIEW OF CURRENT PRACTICES AND FUTURE POTENTIALS. 6(3) :707–721. Number : 3.
- [32] Farid Kadri, Kahina Abdennbi, Sondès Chaabane, and Issam Nouaouri. Approche basée sur les réseaux de neurones récurrents pour la prévision de la demande de soins aux urgences hospitalières.
- [33] Zeynep Hilal Kilimci, A. Okay Akyuz, Mitat Uysal, Selim Akyokus, M. Ozan Uysal, Berna Atak Bulbul, and Mehmet Ali Ekmis. An improved demand forecasting model using deep learning approach and proposed decision integration strategy for supply chain. 2019(1) :9067367. _eprint : <https://onlinelibrary.wiley.com/doi/pdf/10.1155/2019/9067367>.
- [34] Arindam Mukherjee. Machine learning techniques in supply chain management.

- [35] NumPy Contributors. Numpy, 2024. Consulté le 30 juin 2024.
- [36] pandas Development Team. pandas, 2024. Consulté le 30 juin 2024.
- [37] Transition Technologies PSC. Comment l'ia va-t-elle changer la prévision de la demande dans la chaîne d'approvisionnement, 2024. Consulté : 2024-06-21.
- [38] Shuyun Ren, Hau-Ling Chan, and Tana Siqin. Demand forecasting in retail operations for fashionable products : methods, practices, and real case study. Annals of Operations Research, 2019.
- [39] Benkachcha Said, Jamal Benhra, and Hicham El Hassani. PREVISION DE LA DEMANDE DANS LA CHAINE LOGISTIQUE COMBINAISON d'UN MODELE CAUSAL AVEC LES ALGORITHMES GENETIQUES.
- [40] scikit-learn Contributors. scikit-learn, 2024. Consulté le 30 juin 2024.
- [41] Mahya Seyedan, Fereshteh Mafakheri, and Chun Wang. Order-up-to-level inventory optimization model using time-series demand forecasting with ensemble deep learning. 3 :100024, 2023.
- [42] Slimstock. Planification des approvisionnements. <https://www.slimstock.com/fr/blog/planification-approvisionnements/>, 2024. Consulté le 6 juillet 2024.
- [43] Transition Technologies PSC. Comment l'ia va-t-elle changer la prévision de la demande dans la chaîne d'approvisionnement ? Transition Technologies PSC Blog, 2023. Consulté le 21 juin 2024.
- [44] Mariem Trojet. Planification d'une chaîne logistique : approche par satisfaction de contraintes dynamiques.
- [45] Neha Vishwakarma. What is adam optimizer, September 2023.
- [46] Chien-Chih Wang, Chun-Hua Chien, and Amy J. C. Trappey. On the application of ARIMA and LSTM to predict order demand based on short lead time and on-time delivery requirements. 9(7) :1157, 2021. Number : 7 Publisher : Multidisciplinary Digital Publishing Institute.
- [47] Wikipedia contributors. Matplotlib. <https://en.wikipedia.org/wiki/Matplotlib>, 2024. Consulté le 30 juin 2024.

Résumé

Dans ce mémoire, nous avons développé un modèle innovant intégrant l'intelligence artificielle et l'apprentissage automatique, notamment à travers l'utilisation de réseaux neuronaux pour les séries temporelles. Après une analyse approfondie des données incluant le nettoyage et la normalisation, notre approche ensembliste combine les résultats de réseaux neuronaux 1D-CNN et GRU. Ces résultats servent d'entrées à un modèle XGBoost pour prédire la demande logistique, montrant une amélioration significative de la précision des prévisions. Ce projet de Master vise ainsi à optimiser la gestion des stocks et à ouvrir de nouvelles perspectives pour l'avenir de la gestion logistique.

Mots clés : Prédiction de la demande ; Chaîne logistique ; Séries chronologiques ; Intelligence artificielle ; Apprentissage automatique ; Apprentissage profond ; 1D-CNN ; GRU ; XGBoost.

Abstract

In this document, we have developed an innovative model that integrates artificial intelligence and machine learning, leveraging neural networks for time series analysis. Following a thorough data analysis involving cleaning and normalization, our ensemble approach combines outputs from 1D-CNN and GRU neural networks. These results are then input into an XGBoost model to predict logistic demand, resulting in significantly improved forecast accuracy. This Master's project aims to optimize inventory management and explore new avenues for the future of logistics management.

Key words : Demand forecasting ; Supply chain ; Time series ; Artificial intelligence ; Machine Learning ; Deep Learning ; 1D-CNN ; GRU ; XGBoost.