

الجمهورية الجزائرية الديمقراطية الشعبية

وزارة التعليم العالي والبحث العلمي

جامعة سعيدة د. مولاي الطاهر

كلية الرياضيات و الإعلام الآلي و الاتصالات السلكية و اللاسلكية

قسم: الإعلام الآلي



Mémoire de Master en informatique Spécialité : MICR

T h è m e

START UP : DATA ANALYSIS & IA

▪ **Présenté par :**

GRINE Elmountaser billah

BAHLOULI Abdelkader

SELLAOUI Oussama

▪ **Dirigé par :**

Dr. RAHMANI MOHAMED Elhadi

Dr. BOUARARA HADJ AHMAD

Année universitaire



2024-2025

Table des matières

Table des matières	3
Dédicace	6
Remerciements :	7
Listes des tableaux :	8
Listes des Figures :	9
Listes des Abréviations :	10
Introduction générale :	11
Partie I : Cadre théorique et état de l'art	14
1.1. Introduction :	15
1.2. L'analyse des ventes : définition et enjeux :	15
1.2.1. Objectifs de l'analyse des ventes :	15
1.2.2. Types de données utilisées :	15
1.3. Séries temporelles et prévision :	16
1.3.1. Définition d'une série temporelle :	16
1.3.2. Composantes d'une série temporelle :	16
1.4. Intelligence artificielle, apprentissage automatique et profond :	16
1.4.1. Intelligence artificielle (IA) :	16
1.4.2. Apprentissage automatique (Machine Learning) :	16
1.4.3. Apprentissage profond (Deep Learning) :	17
1.5. Modèles de prévision des ventes :	17
1.5.1. MLP (Multi-Layer Perceptron) :	17
1.5.2. Réseaux convolutifs (CNN) :	17
1.5.3. Réseaux LSTM :	17
1.5.4. Architectures hybrides CNN-LSTM :	18
1.6. Chatbots intelligents et modèles de langage :	18
1.6.1. Évolution des chatbots :	18
1.6.2. Modèles utilisés :	18
1.7. Technologies utilisées dans le projet :	18
1.8. Conclusion de la Partie I :	19
Partie II : Conception et Réalisation d'une Plateforme d'Analyse et de Prévision des Ventes avec Flask, MLP, CNN, LSTM, CNN-LSTM et un Chatbot IA.....	20

1. Introduction :	21
2. Architecture Générale de la Plateforme et diagrammes UML de Conception :	21
2.1. Architecture Générale de la plate-forme :	21
2.2 Diagrammes de Cas d'Utilisation et Sequences :	25
2.3 Diagramme de classe :	31
3. Choix Technologiques :	32
4. Fonctionnalités de la Plateforme :	32
4.1 Téléversement des Fichiers :	32
L'utilisateur importe un fichier CSV contenant les historiques de vente. Le système vérifie la validité du format, l'enregistre et affiche un aperçu des données.	32
4.2 Analyse Exploratoire des Données :	32
4.3 Prédiction par Modèles IA :	32
4.4 Interaction avec le Chatbot IA :	33
5. Processus de Développement :	34
5.1 Prétraitement des Données :	34
5.2 Construction des Modèles :	34
5.3 Intégration dans Flask :	34
5.4 Visualisation des Résultats :	35
6. Sécurité, Session et Convivialité :	35
7. Exemple d'Utilisation :	35
8. Difficultés Techniques Rencontrées :	35
9. Conclusion de la Partie II :	36
Partie III : Résultats et Interprétations	37
1. Introduction :	38
2. Méthodologie d'Évaluation :	38
2.1 Métrique utilisée : RMSE :	38
3. Résultats Expérimentaux :	39
4. Analyse et Interprétation :	39
4.1 MLP : un modèle simple mais efficace :	39
4.2 CNN : extraction locale des motifs :	39
4.3 CNN-LSTM : puissance mais surapprentissage léger :	40
5. Visualisation et Interprétation Graphique :	40

6. Synthèse Comparative :.....	42
7. Rôle du Chatbot IA dans l'Analyse :.....	43
8. Conclusion de partie III :.....	43
Conclusion Générale et Perspectives :.....	44
Conclusion Générale :.....	45
Perspectives :.....	46
Mot de la fin :.....	47
Bibliographie :	48
Annexe :.....	50
Annexe A – Extrait du jeu de données utilisé :.....	50
Annexe B – Organisation des fichiers de l'application :	51
Annexe C – Environnement de développement :	52
Annexe D – Interfaces de la plateforme Flask :.....	53
ملخص.....	54
Résumé.....	54
Abstract.....	54

Dédicace

Au nom d'Allah, le Tout Miséricordieux, le Très Miséricordieux

Nous dédions ce modeste travail, avant tout, à Allah Tout-Puissant, qui nous a accordé le savoir, la force et la persévérance nécessaires pour mener à bien cette entreprise.

À nos pères bien-aimés, vous avez été une source constante de soutien, d'encouragement et d'inspiration. Pour vos efforts inlassables de jour comme de nuit, vos invocations et vos sacrifices pour notre éducation et notre bien-être,

À la lumière de nos jours, la source de nos efforts, la flamme de nos cœurs, nos vies et notre bonheur, nos chères mères que nous chérissons tant. Votre amour inconditionnel, votre patience et vos prières ont été notre force motrice.

Puisse ce travail être un moyen d'atteindre la satisfaction d'Allah Tout-Puissant et de réaliser les vœux et les aspirations que vous avez longtemps exprimés. Qu'il soit le témoignage de votre soutien indéfectible et le reflet des valeurs islamiques que vous nous avez transmises.

Nous implorons Allah d'accepter cet humble effort et d'en faire une source de bienfait pour l'humanité.

Remerciements :

Nous tenons tout d'abord à exprimer notre profonde gratitude à **Dieu Tout-Puissant**, pour nous avoir accordé la force, la patience et la persévérance nécessaires à la réalisation de ce mémoire.

Nos remerciements les plus sincères vont à notre **encadrant universitaire**, **RAHMANI MOHAMED EL HADI** et **BOUARARA HADJ AHMAD**, pour sa disponibilité, ses conseils éclairés, son suivi rigoureux et son soutien constant tout au long de ce travail. Son expertise et ses observations pertinentes ont été déterminantes dans l'aboutissement de notre projet.

Nous exprimons également notre reconnaissance à l'ensemble du **corps enseignant de UNIVERSITE DE SAIDA**, pour la qualité de l'enseignement dispensé, la richesse des connaissances transmises, et l'accompagnement tout au long de notre formation.

Nos remerciements s'adressent aussi à **notre jury de soutenance**, pour avoir accepté d'évaluer notre travail et pour les remarques constructives qui permettront d'enrichir davantage notre réflexion.

Un grand merci à **nos familles**, pour leur amour, leur soutien moral et leur confiance inébranlable. Leur présence a été une source de réconfort et de motivation dans les moments les plus exigeants.

Enfin, nous remercions **nos amis et collègues**, avec qui nous avons partagé entraide, encouragements et esprit d'équipe. Votre soutien tout au long de cette aventure universitaire a été précieux.

Listes des tableaux :

<u>Tableau 1 - Technologies utilisées</u>	24
<u>Tableau 2 - choix technologiques</u>	27
<u>Tableau 3 - Construction des Modèles</u>	29
<u>Tableau 4 - Résultats Expérimentaux</u>	34
<u>Tableau 5 - Synthese comparative</u>	37
<u>Tableau 6 - jeu de données</u>	44
<u>Tableau 7 - Organisation des fichiers de l'application</u>	45
<u>Tableau 8 - Environnement de développement</u>	46

Listes des Figures :

<u>Figure 1- diagramme de cas utilisation pour acces a plateforme</u>	25
<u>Figure 2 - diagramme de sequence acces a plateforme</u>	26
<u>Figure 3- diagramme de cas utilisation d'interaction avec chatbot</u>	27
<u>Figure 4-diagramme de sequence d'interaction avec chatbot</u>	27
<u>Figure 5- diagramme de cas utilisation analyse des données via dashboard</u>	28
<u>Figure 6- diagramme de sequence analyse des données via dashboard</u>	29
<u>Figure 7- diagramme de cas utilisation de prédiction des ventes</u>	30
<u>Figure 8- diagramme de séquence de prédiction des ventes</u>	30
<u>Figure 9- diagramme de classe généralise</u>	31
<u>Figure 10- relation pour calcul RMSE</u>	38
<u>Figure 11-RMSE - MLP</u>	41
<u>Figure 12-RMSE - CNN</u>	41
<u>Figure 13-RMSE –(CNN-LSTM)</u>	42
<u>Figure 14-Interface d’accueil</u>	53
<u>Figure 15-Interface d’upload et de dashboard</u>	54
<u>Figure 16-resultat de prediction CNN-LSTM</u>	55
<u>Figure 17-interface de chatbot IA</u>	56
Figure 18:mlp.....	22
Figure 39+1:CNN (Réseau de Neurones Convolutif)	23
Figure 49:LSTM (Long Short-Term Memory)	24
Figure 58: CNN-LSTM (Architecture Hybride)	24

Listes des Abréviations :

IA Intelligence Artificielle

MLP Multi-Layer Perceptron

CNN Convolutional Neural Network

LSTM Long Short-Term Memory

CNN-LSTM Modèle hybride combinant CNN et LSTM

RMSE Root Mean Squared Error

CSV Comma-Separated Values

API Application Programming Interface

UI User Interface

EDA Exploratory Data Analysis

DL Deep Learning

ML Machine Learning

ReLU Rectified Linear Unit

Flask Framework Python léger pour le développement web

Groq Fournisseur d'API pour modèles d'IA (comme LLaMA 3)

Introduction générale :

L'évolution rapide du numérique et l'avènement de l'intelligence artificielle ont profondément transformé les pratiques commerciales et la manière dont les entreprises prennent leurs décisions stratégiques. Dans un contexte où la compétitivité repose de plus en plus sur l'analyse des données, la capacité à anticiper les ventes et à comprendre le comportement des consommateurs est devenue un atout majeur pour les entreprises de toutes tailles.

Les technologies modernes telles que le *machine learning*, les réseaux de neurones profonds (CNN, LSTM) et les plateformes web interactives comme Flask permettent aujourd'hui de créer des outils puissants capables d'analyser des volumes importants de données en temps réel. L'intégration d'un assistant conversationnel intelligent (chatbot) basé sur des modèles de langage récents comme LLaMA 3 via l'API Groq renforce davantage l'interactivité et l'accessibilité de ces plateformes, en rendant l'analyse compréhensible même pour des utilisateurs non techniques.

Ce projet s'inscrit dans cette dynamique. Il vise à concevoir et développer une plateforme web complète, capable d'effectuer une analyse poussée des ventes à partir de fichiers CSV, de générer des prévisions fiables à l'aide de modèles hybrides CNN-LSTM, et de proposer une interface intuitive intégrant un chatbot IA pouvant répondre aux questions de l'utilisateur sur ses propres données.

Problématique :

Dans un marché toujours plus complexe et concurrentiel, les entreprises doivent relever plusieurs défis :

- Comment exploiter efficacement les données historiques de ventes ?
- Quels modèles d'IA sont les plus adaptés à la prévision de la demande ?
- Comment rendre cette analyse accessible à des utilisateurs non experts ?
- De quelle manière peut-on intégrer une assistance intelligente pour interpréter les résultats automatiquement ?

Ces interrogations soulignent l'importance de créer une solution unifiée, à la fois performante, explicative et intuitive.

Objectifs du projet :

L'objectif principal de ce projet est de développer une **plateforme intelligente d'analyse et de prévision des ventes**. Plus précisément, il s'agit de :

- Concevoir une application web à l'aide du framework **Flask**.
- Intégrer un moteur d'analyse basé sur des **modèles de deep learning**, notamment **CNN**, **LSTM** et **CNN-LSTM**.
- Permettre le **téléchargement de fichiers CSV** et leur traitement automatique.
- Afficher des visualisations pertinentes (courbes de ventes, heatmaps, etc.).
- Fournir des **prévisions fiables** à court ou moyen terme.
- Intégrer un **chatbot IA** (via **LLaMA 3 avec l'API Groq**) pour répondre aux questions de l'utilisateur en langage naturel.

Méthodologie adoptée :

Pour atteindre ces objectifs, une approche méthodologique en trois phases a été adoptée :

1. **Recherche documentaire** : analyse de l'existant, étude des modèles CNN, LSTM et hybrides, choix des outils technologiques.

2. **Développement technique** : implémentation des modèles, création de l'interface web, intégration du chatbot IA.
3. **Évaluation** : tests sur des données réelles, comparaison des performances (à l'aide d'indicateurs comme le RMSE), analyse des résultats.

Structure du mémoire :

Ce mémoire est structuré en trois grandes parties :

- **Partie I** : Présente les fondements théoriques liés à l'analyse des ventes, à l'IA et aux modèles de deep learning utilisés.
- **Partie II** : Décrit le processus de conception et de mise en œuvre de la plateforme.
- **Partie III** : Expose les résultats obtenus, compare les performances des modèles, et interprète les prédictions fournies.

Enfin, une conclusion générale résume l'ensemble du travail, souligne les limites du projet et propose des perspectives d'amélioration.

Partie I : Cadre théorique et état de l'art

1.1. Introduction :

L'analyse prédictive des ventes s'impose aujourd'hui comme une discipline clé dans le pilotage stratégique des entreprises. Grâce à l'essor du big data, de l'intelligence artificielle (IA) et des réseaux de neurones profonds (Deep Learning), il est désormais possible de modéliser les tendances de la demande et d'anticiper les fluctuations du marché avec une précision croissante « (Choi et Varian, 2012 ; Fischer & Krauss, 2018). »

Cette première partie présente les fondements conceptuels et les approches scientifiques mobilisées dans ce projet. Elle explore successivement les principes de l'analyse des ventes, les définitions clés en intelligence artificielle, puis les différents modèles utilisés dans la prédiction des séries temporelles, notamment le MLP, le CNN, le LSTM, et les architectures hybrides CNN-LSTM.

1.2. L'analyse des ventes : définition et enjeux :

1.2.1. Objectifs de l'analyse des ventes :

L'analyse des ventes vise à :

- Anticiper les volumes futurs de ventes.
- Identifier les produits ou segments les plus performants.
- Optimiser la gestion des stocks et des ressources.
- Comprendre les cycles de saisonnalité.
- Détecter les anomalies dans la demande.

1.2.2. Types de données utilisées :

Les données commerciales exploitables incluent :

- Les ventes par jour, semaine ou mois.
- Les métadonnées des produits (prix, catégorie, région).
- Les informations client (profil, historique d'achat).

- Les facteurs exogènes (saisons, fêtes, promotions, événements).

1.3. Séries temporelles et prévision :

1.3.1. Définition d'une série temporelle :

Une série temporelle est une suite ordonnée de valeurs mesurées à des intervalles temporels constants. Elle permet d'observer et de prédire l'évolution d'un phénomène au fil du temps.

1.3.2. Composantes d'une série temporelle :

- **Tendance (Trend)** : évolution globale à long terme.
- **Saisonnalité (Seasonality)** : fluctuations périodiques.
- **Cycles (Cycles)** : mouvements irréguliers liés à l'environnement.
- **Résidus (Noise)** : variations aléatoires imprévisibles.

1.4. Intelligence artificielle, apprentissage automatique et profond :

1.4.1. Intelligence artificielle (IA) :

L'IA désigne un ensemble de techniques visant à simuler l'intelligence humaine dans des systèmes informatiques (Russell & Norvig, 2016).

1.4.2. Apprentissage automatique (Machine Learning) :

Le ML permet aux systèmes d'apprendre à partir de données :

- **Supervisé** : avec variables cibles (ex. régression).
- **Non-supervisé** : sans cibles (ex. clustering).
- **Renforcement** : interaction avec un environnement.

1.4.3. Apprentissage profond (Deep Learning) :

Le deep learning repose sur des réseaux de neurones à plusieurs couches. Il est adapté aux données massives et séquentielles comme les séries temporelles « (Goodfellow et al., 2016). »

1.5. Modèles de prévision des ventes :

1.5.1. MLP (Multi-Layer Perceptron) :

Le MLP est une architecture basique mais efficace pour la classification et la régression sur données tabulaires. Toutefois, il traite difficilement les dépendances temporelles.

« *Bontempi et al. (2013)* ont utilisé les MLP pour prévoir des flux financiers, avec des résultats modestes comparés aux modèles récurrents. »

1.5.2. Réseaux convolutifs (CNN) :

Les CNN sont conçus pour extraire automatiquement des motifs dans les données « « (Goodfellow et al., 2016) ». Appliqués aux séries temporelles, ils détectent des régularités locales comme la saisonnalité « (Borovykh et al., 2017). »

« Par exemple, *Zhao et al. (2017)* ont utilisé un CNN pour prédire les ventes hebdomadaires d'un réseau de magasins, en extrayant des motifs locaux significatifs. »

1.5.3. Réseaux LSTM :

Le Long Short-Term Memory (LSTM) est un réseau récurrent qui résout le problème de gradient évanescent des RNN classiques. Il retient l'information pertinente sur de longues séquences « (Hochreiter & Schmidhuber, 1997). »

Dans le domaine économique, « *Fischer & Krauss (2018)* ont démontré la supériorité des LSTM sur les méthodes traditionnelles pour prédire les cours boursiers. »

1.5.4. Architectures hybrides CNN-LSTM :

La combinaison CNN-LSTM permet :

- D'extraire des caractéristiques locales (via CNN),
- Puis de modéliser la dépendance temporelle globale (via LSTM).

« *Cao et al. (2020)* ont montré que les modèles CNN-LSTM surpassent les architectures simples dans la prédiction de la consommation électrique, des ventes, ou des cours de bourse. »

1.6. Chatbots intelligents et modèles de langage :

1.6.1. Évolution des chatbots :

Les premiers chatbots utilisaient des règles simples. Aujourd'hui, les modèles de langage de grande taille (LLMs) comme GPT ou LLaMA offrent des capacités conversationnelles de niveau humain « (Brown et al., 2020). »

1.6.2. Modèles utilisés :

Dans ce projet, nous utilisons LLaMA 3 via l'API Groq pour intégrer un chatbot capable d'interpréter les résultats d'analyse, d'expliquer les prévisions, et de répondre aux questions liées aux données.

1.7. Technologies utilisées dans le projet :

Technologie	Rôle
Python	Langage de développement principal
Flask	Framework web léger pour l'interface
Pandas / NumPy	Traitement des données CSV
Matplotlib / Seaborn / Plotly	Visualisation des données
TensorFlow / Keras	Entraînement des modèles MLP, CNN, LSTM
LLaMA 3 (Groq API)	Intégration du chatbot intelligent
HTML / CSS / Bootstrap	Frontend de l'interface utilisateur

1.8. Conclusion de la Partie I :

Ce premier chapitre a posé les bases théoriques nécessaires pour comprendre les fondements du projet. Il a permis de contextualiser l'utilisation du deep learning dans les systèmes de prévision commerciale, tout en justifiant le choix des modèles CNN, LSTM et hybrides. Dans la deuxième partie, nous passerons à la conception et la réalisation technique de la plateforme.

Partie II : Conception et Réalisation d'une Plateforme d'Analyse et de Prédiction des Ventes avec Flask, MLP, CNN, LSTM, CNN-LSTM et un Chatbot IA

1. Introduction :

Dans un contexte économique marqué par une compétition croissante, les entreprises doivent s'appuyer sur des outils intelligents pour analyser leurs données commerciales et anticiper les tendances de vente. Cette partie décrit la conception et la mise en œuvre d'une plateforme web d'analyse et de prévision des ventes, basée sur le framework Flask. Elle intègre plusieurs modèles d'apprentissage profond, notamment MLP, CNN, LSTM et CNN-LSTM, ainsi qu'un chatbot intelligent via l'API Groq. L'objectif est de fournir une solution accessible, performante et interactive pour la prise de décision commerciale.

2. Architecture Générale de la Plateforme et diagrammes UML de Conception :

2.1. Architecture Générale de la plate-forme :

La plateforme a été pensée de manière modulaire, afin de faciliter la maintenance, l'évolution et l'intégration de nouvelles fonctionnalités. Elle se compose des éléments suivants :

1. Interface Utilisateur (Frontend)

- Développée en **HTML, CSS et Bootstrap**, l'interface offre une expérience utilisateur fluide et ergonomique. Elle permet aux utilisateurs de :
- Naviguer entre les pages (Accueil, Analyse, Prédiction, Chatbot, Contact).
- Télécharger des fichiers CSV contenant les données de ventes.
- Visualiser les résultats sous forme de graphiques interactifs.
- Interagir avec le chatbot intelligent.

2. Serveur Applicatif (Backend)

- Le cœur de la logique applicative est développé en **Python avec le framework Flask**. Ce serveur assure plusieurs rôles :
- La gestion des routes et des requêtes HTTP.
- La réception, le traitement et le stockage temporaire des fichiers CSV dans un dossier uploads/.

- Le déclenchement des modèles IA pour la prédiction.
- L'intégration de l'API du chatbot Groq (LLaMA 3).
- La sécurisation de certaines routes via sessions utilisateurs.

3. Moteur de Prédiction

- Ce module intègre **quatre modèles d'apprentissage automatique** entraînés sur des données temporelles de ventes. Chaque modèle est exposé via une route dédiée dans Flask :

a) MLP (Perceptron Multi-Couche)

- **Architecture :**
- Entrée : Séquence de valeurs numériques (par exemple : ventes passées sur 7 jours).
- Couches cachées : 2 à 3 couches entièrement connectées (Dense), avec activation **ReLU**.
- Couche de sortie : 1 neurone avec activation **linéaire** pour la régression.
- **Fonction :** Prédire la valeur future d'une série à partir de vecteurs de caractéristiques simples.

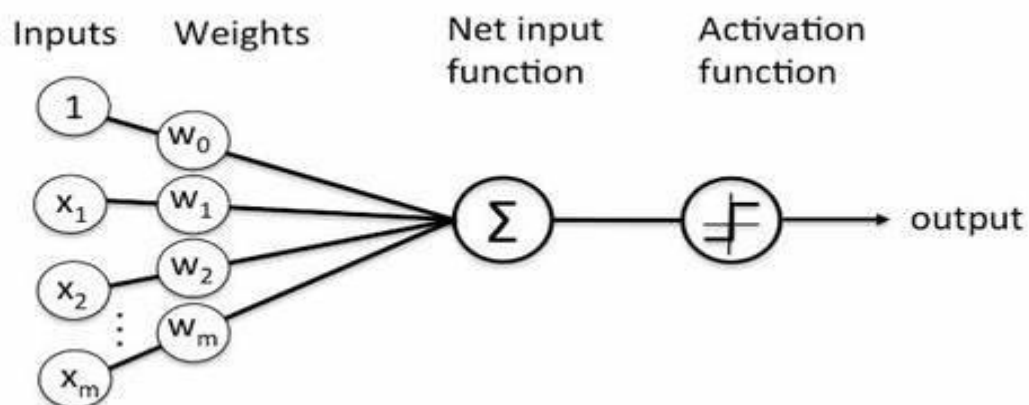


Figure 18:mlp

b) CNN (Réseau de Neurones Convolutif)

- **Architecture :**
- Entrée : Données temporelles remodelées sous forme matricielle.
- Couches convolutionnelles : 1D Conv avec filtres pour détecter des motifs locaux (ex. saisonnalité courte).
- Couches de pooling : réduction de dimensionnalité.
- Dense : couche finale connectée à la sortie pour la prédiction.
- **Avantage :** Efficace pour identifier des motifs récurrents sur des fenêtres temporelles courtes.

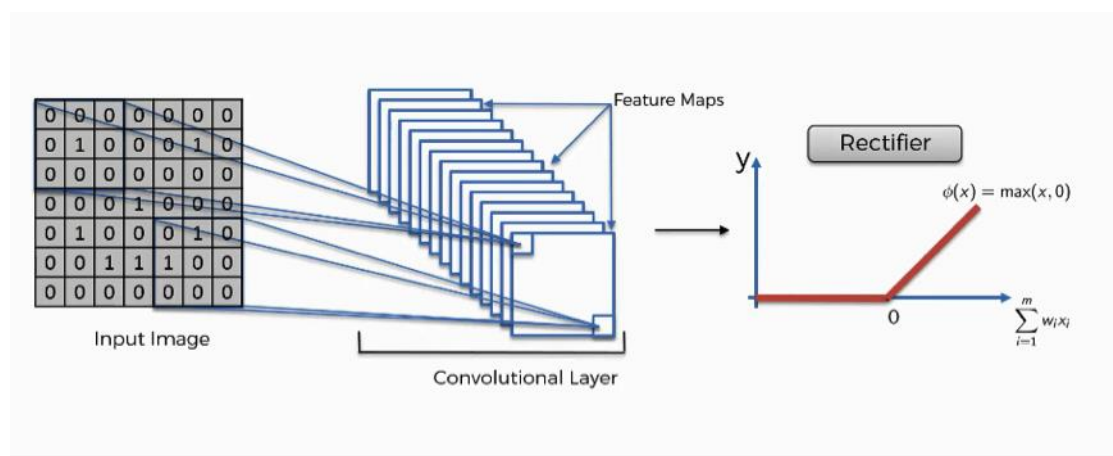


Figure 29+1: CNN (Réseau de Neurones Convolutif)

c) LSTM (Long Short-Term Memory)

- **Architecture :**
- Entrée : Séquences temporelles organisées avec une dimension temporelle.
- Couche LSTM : permet la mémorisation des dépendances à long terme.
- Dropout : pour éviter le surapprentissage.
- Dense : sortie linéaire pour prédiction future.
- **Avantage :** Idéal pour suivre les tendances longues et modéliser les dépendances temporelles complexes.

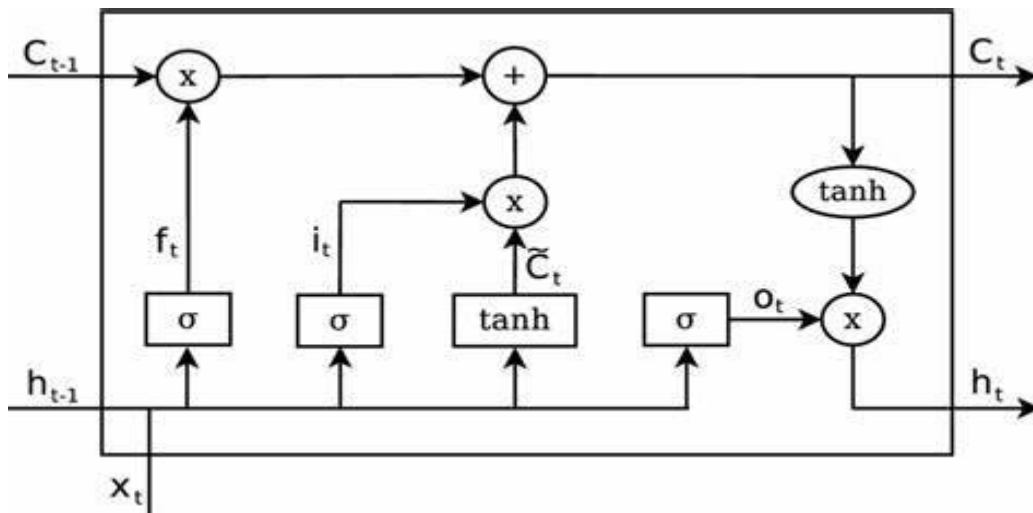


Figure 39: LSTM (Long Short-Term Memory)

d) CNN-LSTM (Architecture Hybride)

- **Architecture :**
- Entrée : Séquences temporelles organisées sous forme matricielle.
- Bloc CNN : couches convolutionnelles pour extraire des motifs locaux.
- Reshape : transformation des sorties CNN pour les adapter à l'entrée LSTM.
- Bloc LSTM : capture de la dynamique temporelle longue.
- Dense : couche finale pour la prédiction.
- **Forces :** Combine les avantages des CNN (détection locale) et LSTM (dépendances globales), pour une prédiction plus précise.

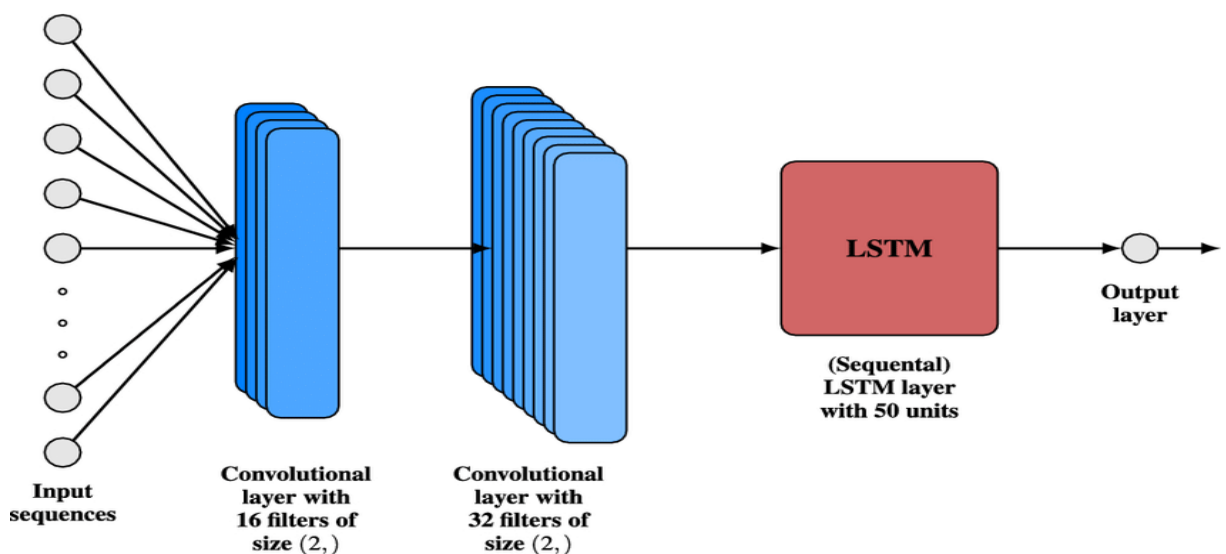


Figure 48: CNN-LSTM (Architecture Hybride)

4. Chatbot Intelligent (IA conversationnelle)

- Intégré via l'**API Groq** et basé sur **LLaMA 3**, ce chatbot :
- Interagit en langage naturel avec l'utilisateur.
- Explique les résultats des modèles.
- Guide l'utilisateur dans l'interprétation des prévisions et la manipulation des données.
- Suggère des stratégies commerciales à partir des tendances détectées.

5. Stockage Temporaire des Données

- Les fichiers CSV uploadés sont stockés temporairement dans le répertoire **uploads/** pour :
- Être prétraités (nettoyage, normalisation).
- Être utilisés comme entrée pour les modèles IA.
- Être éventuellement effacés après traitement pour économiser l'espace disque.

2.2 Diagrammes de Cas d'Utilisation et Sequences :

1. Accès à la plateforme :

L'utilisateur accède à la plateforme en saisissant son adresse web dans un navigateur. Le serveur Flask traite la requête et retourne l'interface d'accueil, sans nécessiter d'authentification. Ce scénario constitue le point d'entrée vers toutes les fonctionnalités de la plateforme, telles que le tableau de bord, l'analyse de données ou le chatbot.

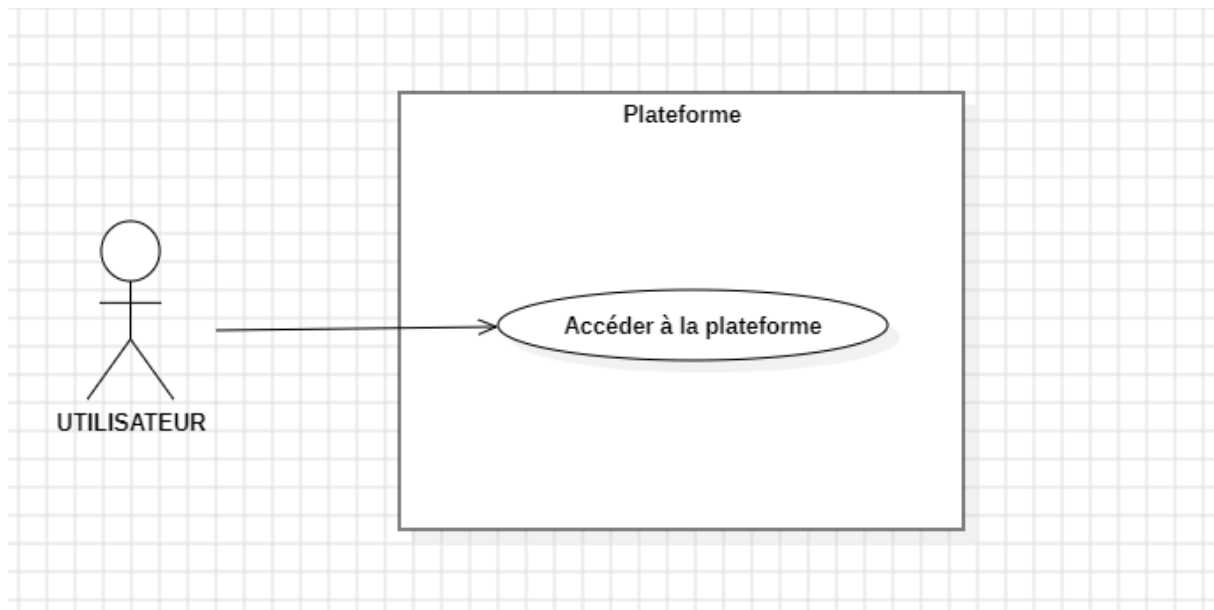


Figure 1- diagramme de cas utilisation pour acces a plateforme

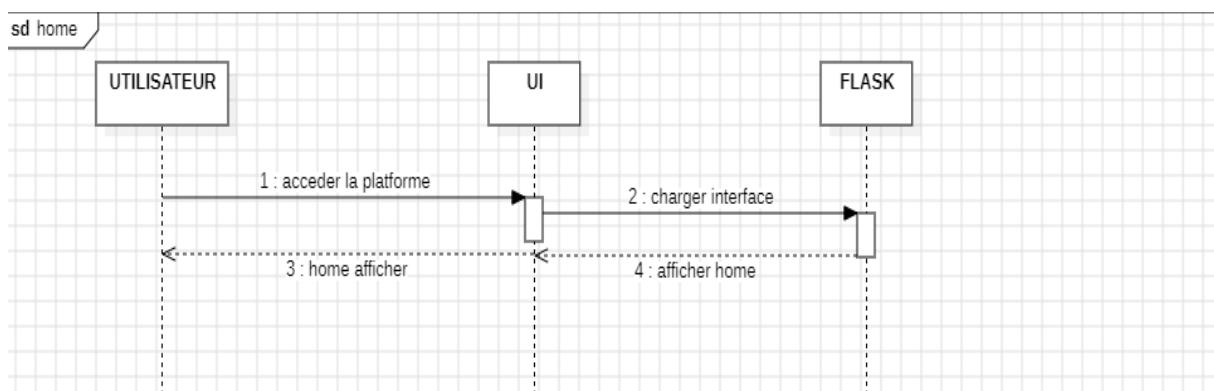


Figure 2 - diagramme de sequence acces a plateforme

2.Interaction avec le Chatbot :

L'utilisateur accède à la plateforme via un navigateur, puis se rend à la section Chatbot. Depuis cette interface, il charge un fichier CSV contenant des données commerciales. Une fois le fichier validé, l'utilisateur peut poser des questions au chatbot en lien avec les données du fichier. Le chatbot IA traite la requête et fournit une réponse intelligente basée sur le contenu analysé du fichier CSV.

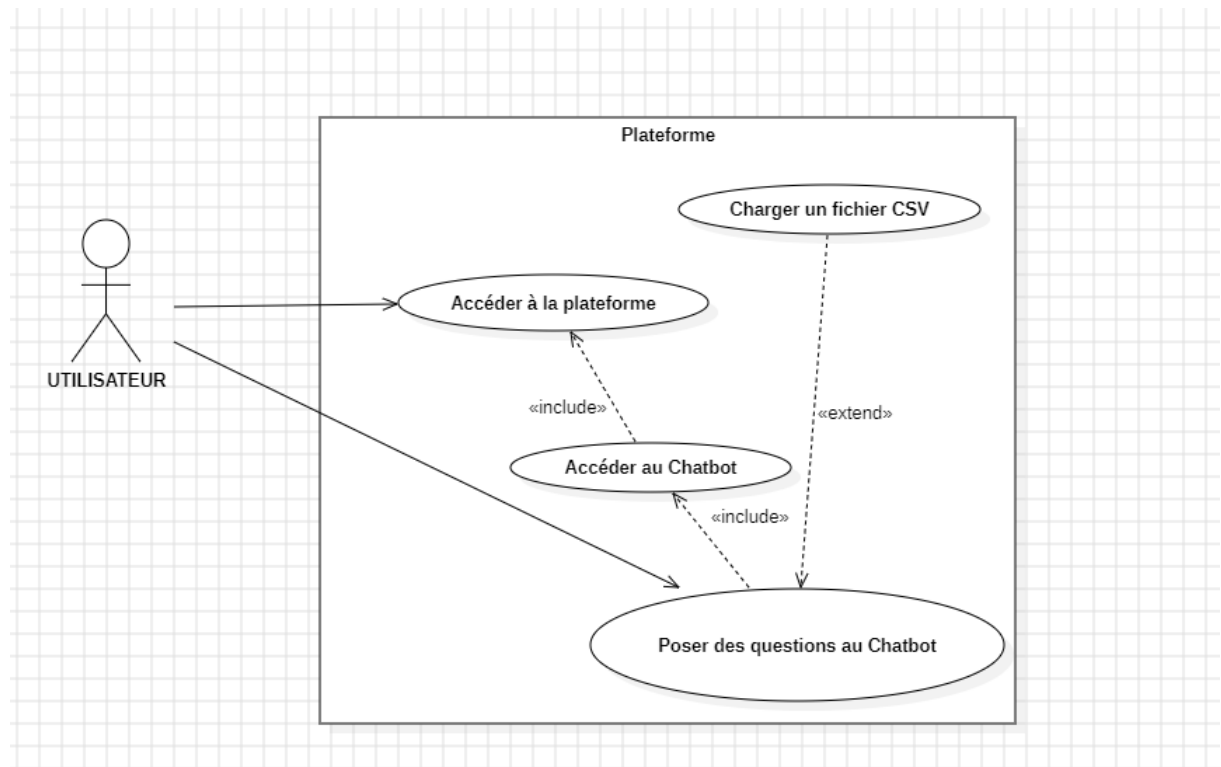


Figure 3- diagramme de cas utilisation d'interaction avec chatbot

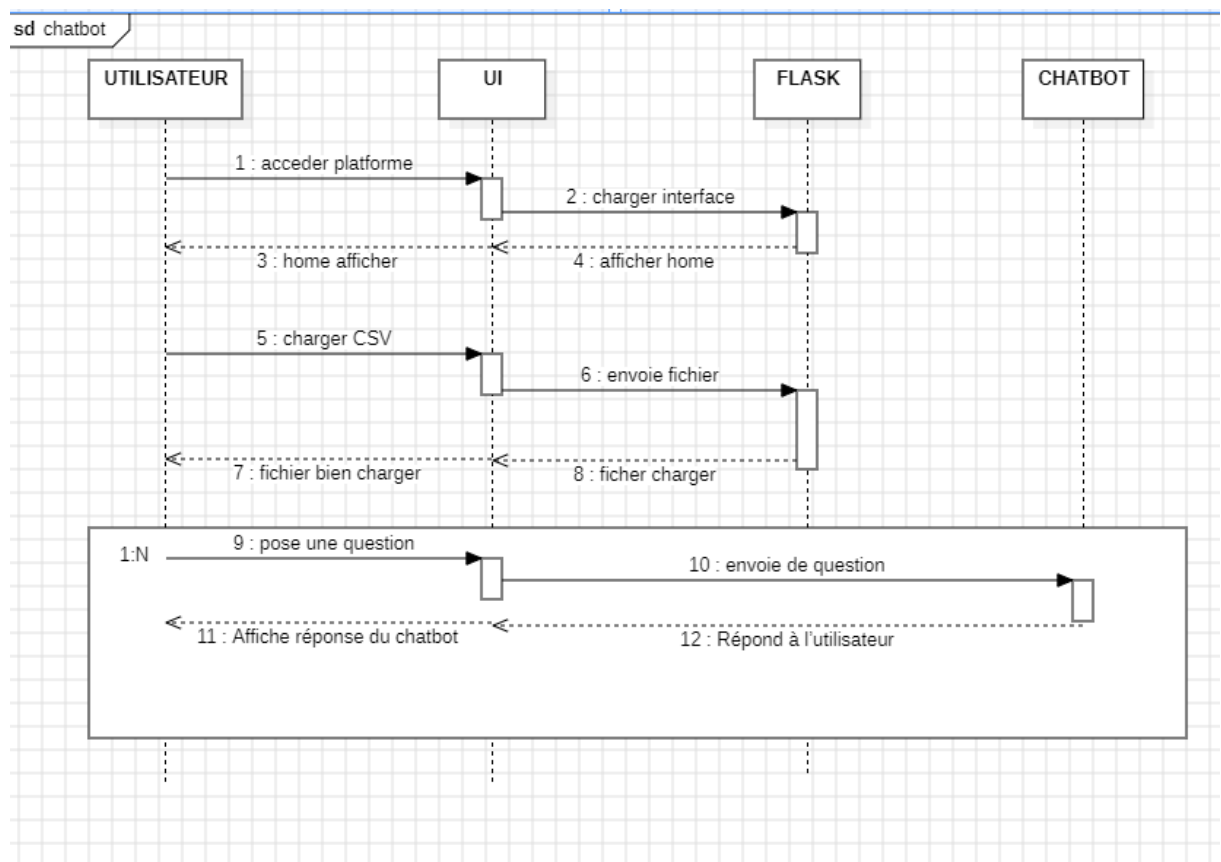


Figure 4-diagramme de sequence d'interaction avec chatbot

3.Visualisation de données via le Dashboard :

L'utilisateur accède à la plateforme depuis un navigateur, puis se rend à la section **Dashboard**. Il y téléverse un fichier CSV contenant des données commerciales. Une fois le fichier chargé et validé, il clique sur le bouton "Analyser". Le serveur traite les données via un module d'analyse, puis affiche des **visualisations interactives** (graphiques, tableaux de statistiques) à l'écran, permettant à l'utilisateur de mieux comprendre le contenu du fichier.

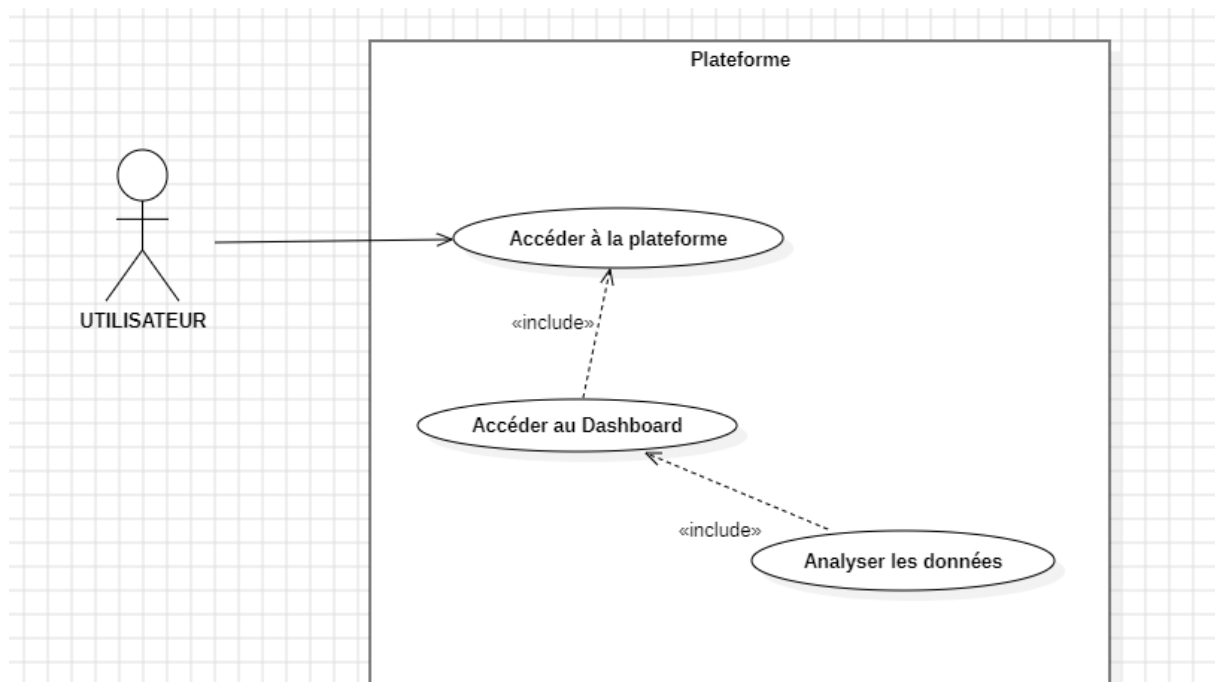


Figure 5- diagramme de cas utilisation analyse des données via dashboard

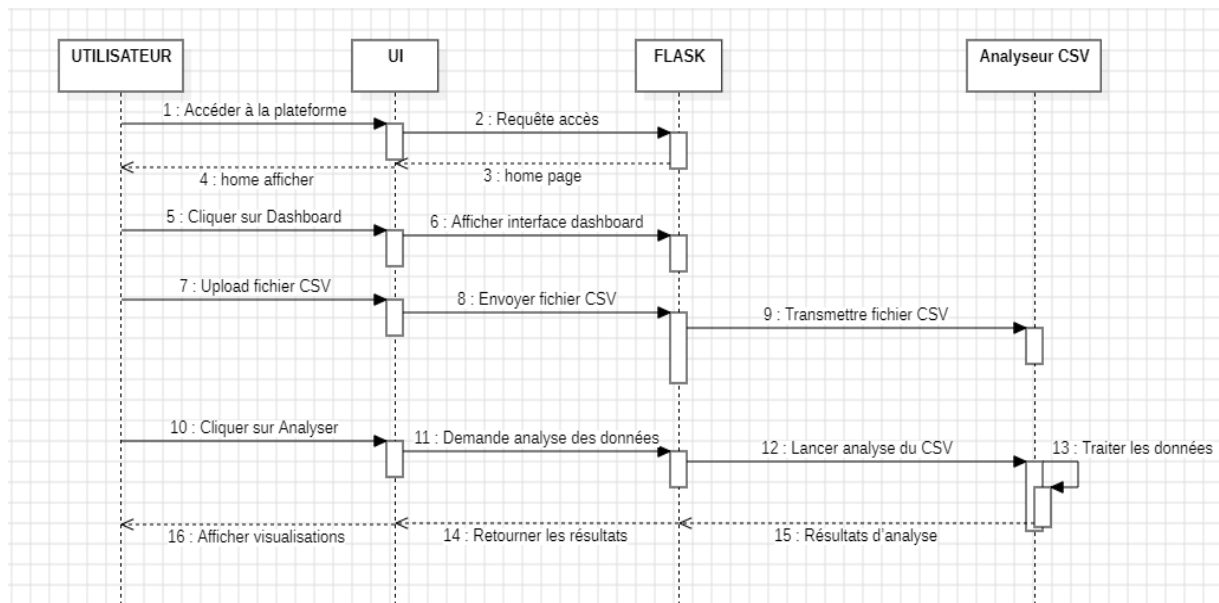


Figure 6- diagramme de sequence analyse des données via dashboard

4.Prédiction des ventes :

L'utilisateur accède à la plateforme via un navigateur, puis se rend à la section **Analyse**. Il y téléverse un fichier CSV contenant des données historiques de ventes. Une fois le fichier validé, il clique sur "Prédire". Le serveur Flask transmet alors les données au modèle **CNN-LSTM**, qui génère des prédictions sur les ventes futures. Ces résultats sont ensuite affichés à l'utilisateur sous forme de courbes ou de tableaux.

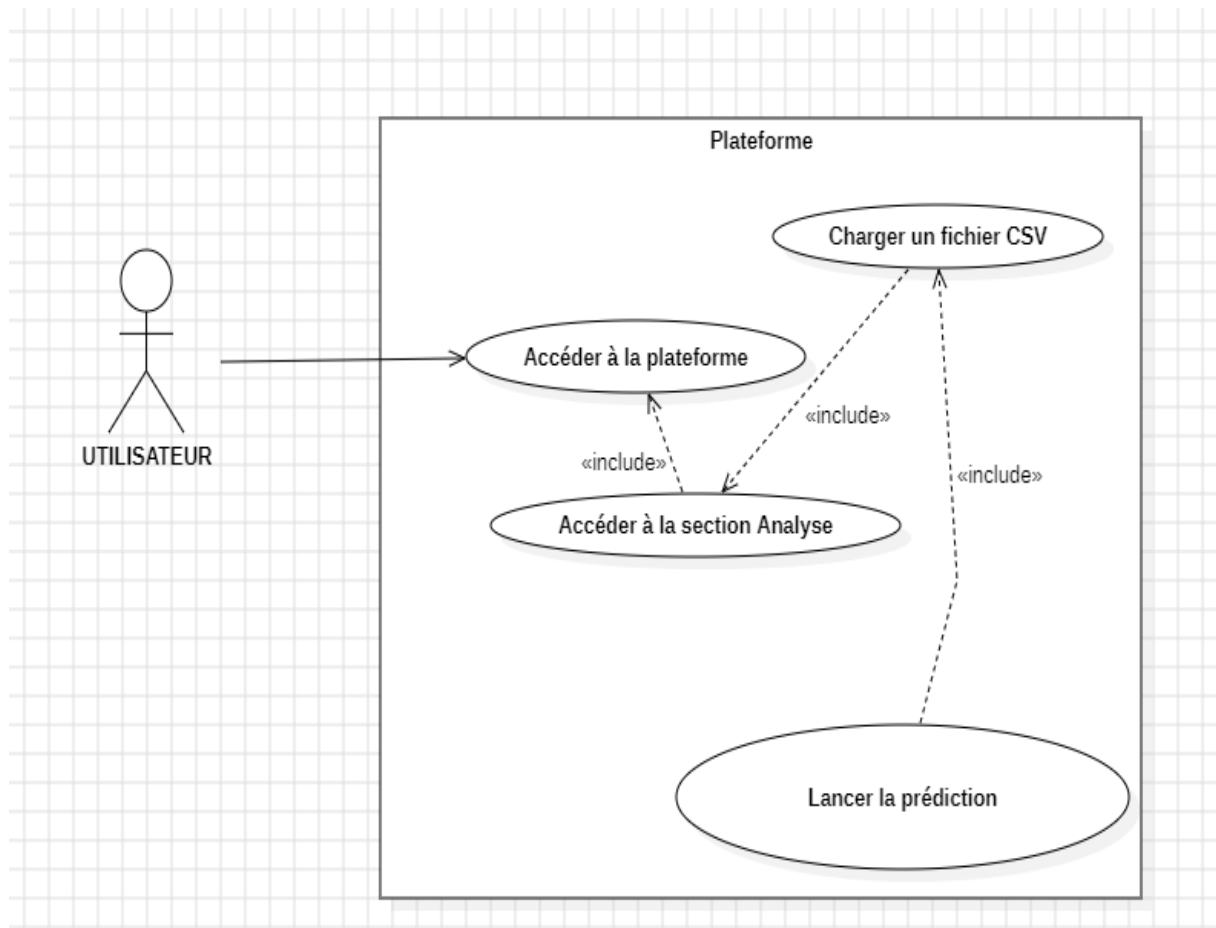


Figure 7- diagramme de cas utilisation de prédiction des ventes

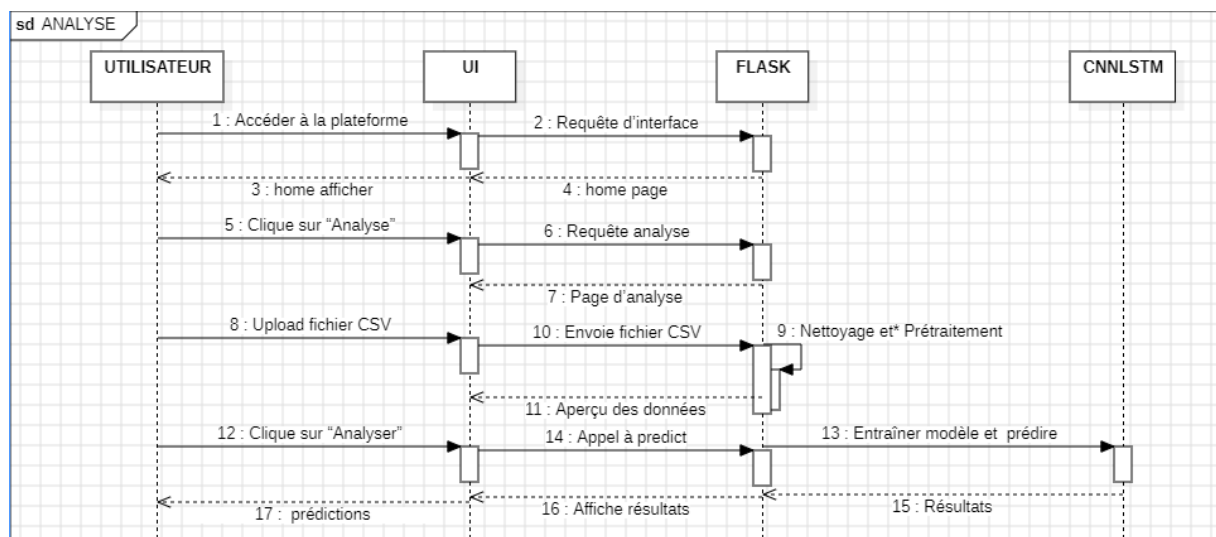


Figure 8- diagramme de séquence de prédiction des ventes

2.3 Diagramme de classe :

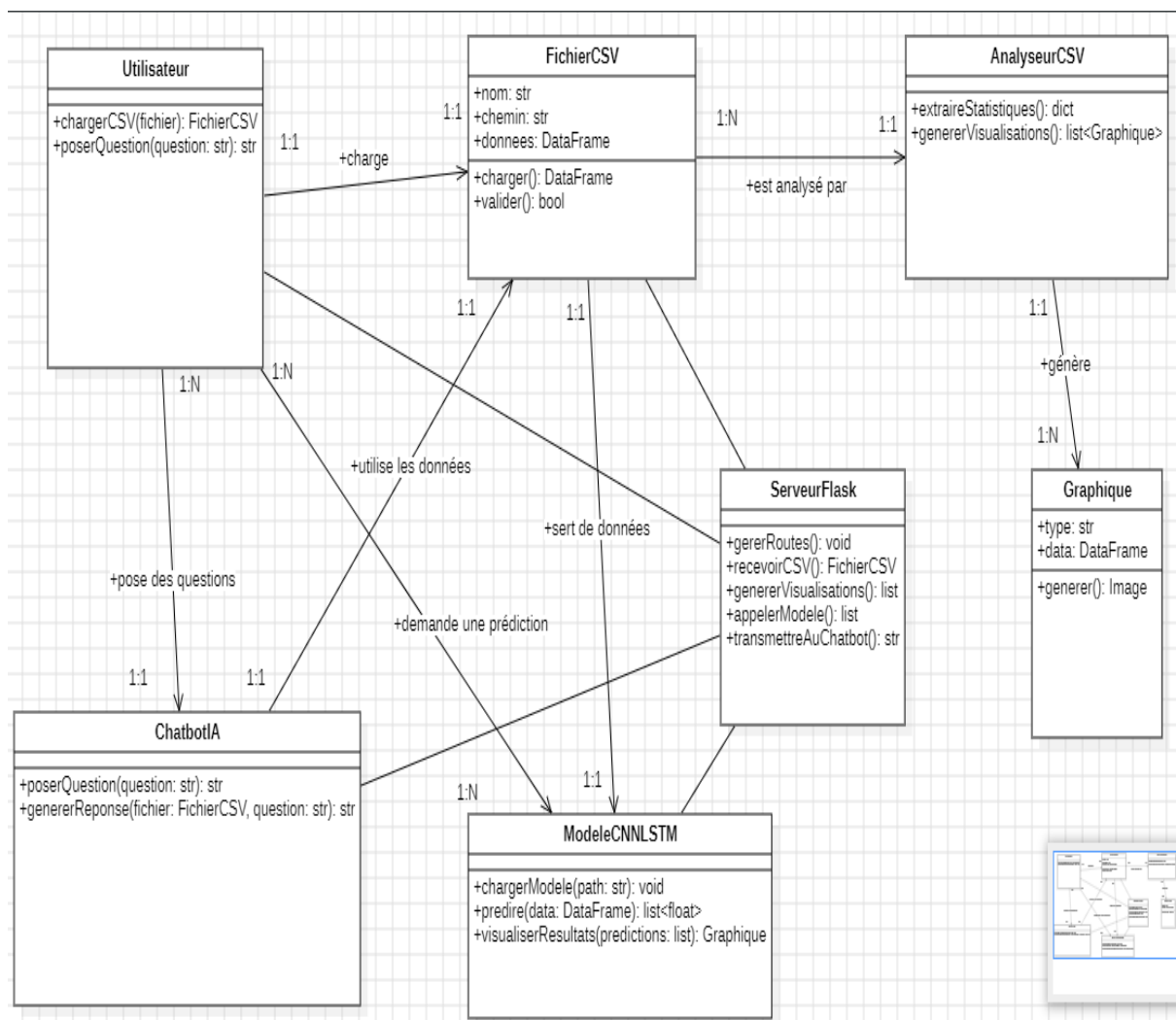


Figure 9- diagramme de classe généralise

3. Choix Technologiques :

Composant	Technologie	Justification
Framework web	Flask	Léger, rapide, facile à intégrer avec les bibliothèques Python
Modèles IA	MLP, CNN, LSTM, CNN-LSTM	Comparaison de performances pour la prévision de séries temporelles
Chatbot	LLaMA 3 via Groq API	Génération de réponses cohérentes et rapides
Bibliothèques	Pandas, NumPy, Keras, TensorFlow, Seaborn, Matplotlib	Traitement de données, entraînement de modèles, visualisation

Tableau 2 - choix technologiques

4. Fonctionnalités de la Plateforme :

4.1 Téléversement des Fichiers :

L'utilisateur importe un fichier CSV contenant les historiques de vente. Le système vérifie la validité du format, l'enregistre et affiche un aperçu des données.

4.2 Analyse Exploratoire des Données :

La plateforme fournit une analyse descriptive complète :

- Statistiques (moyenne, médiane, écart-type, etc.)
- Visualisations temporelles des ventes
- Corrélation entre variables
- Matrice de chaleur et histogrammes

4.3 Prédiction par Modèles IA :

L'utilisateur peut choisir l'un des quatre modèles prédictifs :

a) MLP (Perceptron Multi-Couche) :

Le MLP constitue un modèle de base entièrement connecté. Bien que peu adapté aux séries temporelles brutes, il sert de référence comparative. Il permet de valider la complexité nécessaire aux autres architectures.

b) CNN (Réseau Convolutif) :

Le CNN est utilisé pour détecter des motifs locaux dans les données de séries temporelles. Une couche convolutive 1D permet d'extraire des caractéristiques significatives, suivie d'un réseau dense pour la prédiction.

c) LSTM (Long Short-Term Memory) :

Le LSTM est conçu pour capturer les dépendances temporelles longues, ce qui en fait un choix pertinent pour la prévision de séries chronologiques comme les ventes.

d) CNN-LSTM (Modèle Hybride) :

Ce modèle combine la puissance d'extraction de caractéristiques du CNN avec la capacité de mémoire à long terme du LSTM. Il est le plus sophistiqué et performant de la plateforme.

Chaque modèle est entraîné sur les données utilisateur, puis évalué selon l'erreur quadratique moyenne (RMSE). Les résultats sont visualisés et comparés.

4.4 Interaction avec le Chatbot IA :

Une interface de chat permet à l'utilisateur de poser des questions en langage naturel :

- Interprétation des résultats
- Suggestions d'actions commerciales
- Explications des graphiques
- Résumé des performances des modèles

Le chatbot accède au contexte du fichier CSV chargé et fournit des réponses contextualisées, grâce à la puissance du modèle LLaMA 3 hébergé via l'API Groq.

5. Processus de Développement :

5.1 Prétraitement des Données :

Avant l'entraînement, les données sont nettoyées, normalisées et réorganisées en séquences temporelles adaptées à chaque modèle. Les étapes clés incluent :

- Suppression des valeurs manquantes
- Normalisation MinMax
- Découpage en séquences glissantes pour l'entraînement supervisé

5.2 Construction des Modèles :

Chacun des quatre modèles est encapsulé dans une fonction Python. L'architecture est la suivante :

Modèle	Architecture Résumée
MLP	Dense(128) → Dropout → Dense(64) → Dense(1)
CNN	Conv1D → MaxPooling → Flatten → Dense(64) → Dense(1)
LSTM	LSTM(64) → Dropout → Dense(32) → Dense(1)
CNN-LSTM	Conv1D → MaxPooling → LSTM(64) → Dense(1)

Tableau 3 - Construction des Modèles

5.3 Intégration dans Flask :

Chaque modèle dispose d'une route dédiée (/predict/mlp, /predict/cnn, etc.). Ces routes prennent en entrée le fichier CSV chargé, exécutent le prétraitement, entraînent le modèle, puis retournent :

- Un graphique comparatif (réel vs prédit)
- Le RMSE calculé
- Le chemin de sauvegarde de l'image

5.4 Visualisation des Résultats :

Les prédictions sont accompagnées de graphiques :

- Ventes réelles vs prédites
- Évolution temporelle
- Distribution des erreurs

6. Sécurité, Session et Convivialité :

- Utilisation d'une clé secrète (SECRET_KEY) pour sécuriser les sessions.
- Stockage temporaire des fichiers dans un dossier protégé.
- Historique du chatbot stocké en session pour continuité des échanges.
- Interface intuitive avec navigation claire (Home, Analyse, Dashboard, Chatbot, Services, About us).

7. Exemple d'Utilisation :

Un responsable commercial importe un fichier CSV représentant les ventes journalières d'un magasin. Il consulte les statistiques, identifie une corrélation entre les jours de semaine et les volumes de vente, puis utilise les quatre modèles pour prédire les ventes de la semaine suivante. Le CNN-LSTM fournit les prédictions les plus précises (RMSE minimal). Le chatbot lui suggère d'ajuster les stocks en conséquence.

8. Difficultés Techniques Rencontrées :

- L'hétérogénéité des fichiers CSV utilisateur a nécessité la mise en place d'un prétraitement robuste.
- La gestion mémoire lors de l'entraînement des modèles lourds sur un serveur local a nécessité des optimisations.
- L'intégration de l'API Groq a demandé une gestion soignée des erreurs réseau et de la latence.

9. Conclusion de la Partie II :

La plateforme réalisée offre un outil complet, intelligent et interactif pour l'analyse et la prévision des ventes. L'intégration de quatre modèles (MLP, CNN, LSTM, CNN-LSTM) permet de comparer les performances selon la complexité des données. L'assistant conversationnel enrichit l'expérience utilisateur en rendant l'analyse accessible même aux non-experts. Cette architecture flexible ouvre la voie à de futures évolutions, telles que l'intégration d'un tableau de bord en temps réel ou le déploiement sur le cloud.

Partie III : Résultats et Interprétations

1. Introduction :

Dans cette partie, nous présentons une analyse approfondie des performances des différents modèles d'apprentissage automatique implémentés dans notre plateforme d'analyse des ventes. L'objectif est d'évaluer l'efficacité prédictive des modèles en utilisant des indicateurs de performance adaptés aux séries temporelles, tout en prenant en compte leur complexité et leur capacité de généralisation.

2. Méthodologie d'Évaluation :

2.1 Métrique utilisée : RMSE :

La performance de chaque modèle est évaluée à l'aide de la métrique **RMSE (Root Mean Square Error)**, qui mesure la racine carrée de l'erreur quadratique moyenne entre les valeurs réelles et les valeurs prédites :

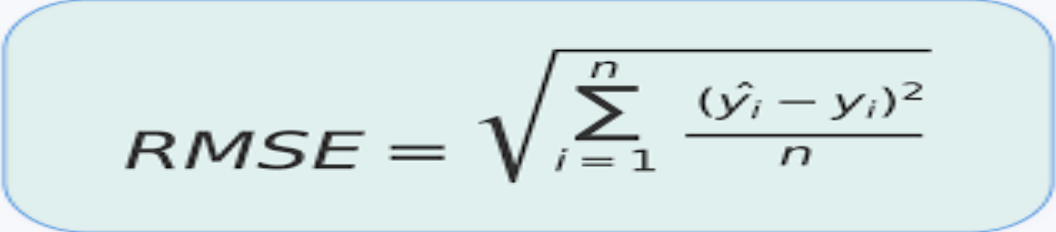

$$RMSE = \sqrt{\sum_{i=1}^n \frac{(\hat{y}_i - y_i)^2}{n}}$$

Figure 10- relation pour calcul RMSE

Où :

- N : nombre total d'observations
- y_i : valeur réelle observée
- $\hat{y}_i$: valeur prédite par le modèle

Le RMSE permet d'avoir une interprétation intuitive de l'erreur de prédiction, exprimée dans la même unité que la variable prédite. Plus il est faible, plus le modèle est précis.

3. Résultats Expérimentaux :

Les résultats obtenus sont présentés dans le tableau suivant :

Modèle	Paramètres	RMSE Entraînement	RMSE Validation
MLP	3,201	18.52928	18.67164
CNN	45,093	18.66020	18.77945
CNN-LSTM	99,979	18.96731	18.96548

Tableau 4 - Résultats Expérimentaux

Ces valeurs permettent une comparaison objective des modèles en termes de **complexité (nombre de paramètres)** et de **précision (erreurs RMSE)**.

4. Analyse et Interprétation :

4.1 MLP : un modèle simple mais efficace :

Le modèle **Multilayer Perceptron (MLP)** présente le **plus faible RMSE** aussi bien pour l'entraînement que pour la validation. Avec un nombre de paramètres très faible (3,201), il offre une **bonne généralisation** sans surapprentissage. Bien qu'il ne prenne pas explicitement en compte la structure temporelle des données, sa simplicité le rend robuste dans des contextes de faible complexité.

✓ **Avantage** : Faible complexité, bon équilibre biais/variance.

✗ **Limite** : Moins performant pour détecter les dépendances temporelles longues.

4.2 CNN : extraction locale des motifs :

Le modèle **Convolutional Neural Network (CNN)**, avec ses 45,093 paramètres, capte efficacement les motifs locaux dans les séquences de ventes. Cependant, il présente un **léger surapprentissage**, car le RMSE en validation est supérieur à celui de l'entraînement. Cela suggère une bonne capacité d'apprentissage, mais une sensibilité aux données non vues.

✓ **Avantage** : Bon pour les motifs saisonniers ou hebdomadaires.

✗ **Limite** : Moins bon pour les tendances longues ou irrégulières.

4.3 CNN-LSTM : puissance mais surapprentissage léger :

Le **modèle hybride CNN-LSTM** est le plus complexe, avec **près de 100,000 paramètres**. Il est conçu pour combiner les atouts des CNN (extraction locale) et des LSTM (dépendances temporelles longues). Toutefois, il présente une **légère divergence entre les RMSE d'entraînement et de validation**, ce qui pourrait indiquer un **début de surapprentissage** malgré des performances correctes.

✓ **Avantage** : Apprend à la fois motifs locaux et relations temporelles complexes.

✗ **Limite** : Plus coûteux à entraîner, peut surapprendre si non régularisé.

5. Visualisation et Interprétation Graphique :

Les graphiques générés dans l'interface Flask permettent de comparer visuellement les prédictions de chaque modèle avec les valeurs réelles. Ils confirment les observations chiffrées :

- **MLP** : suit bien les tendances générales.

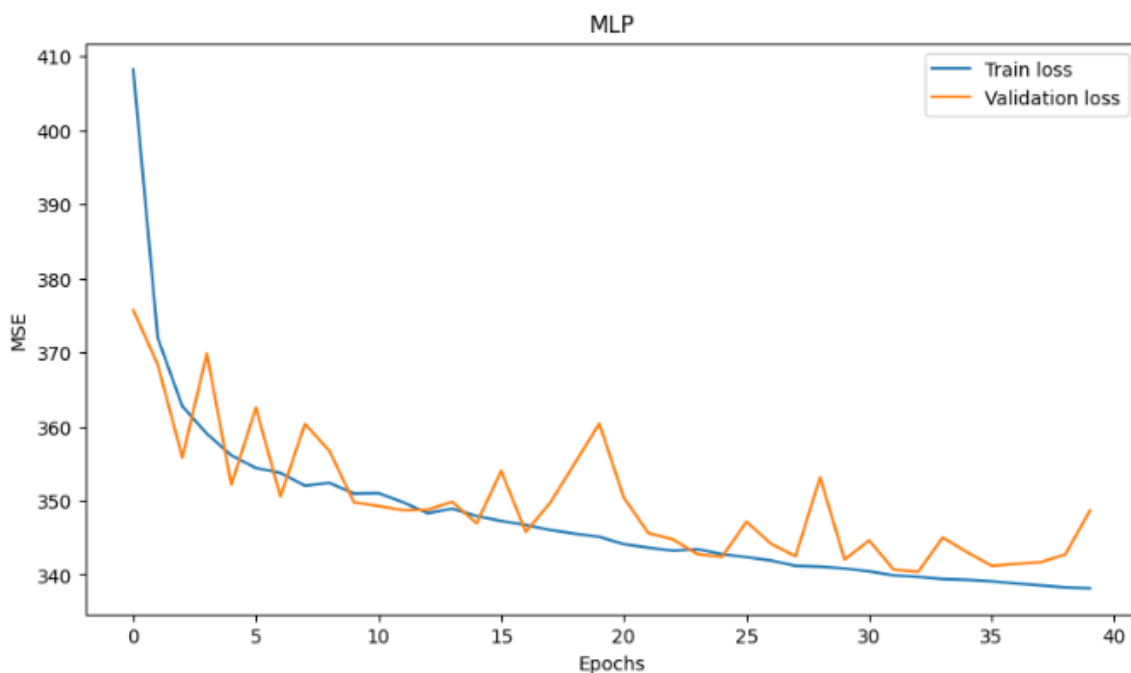


Figure 11-RMSE - MLP

- **CNN** : bonnes performances sur les périodes régulières.

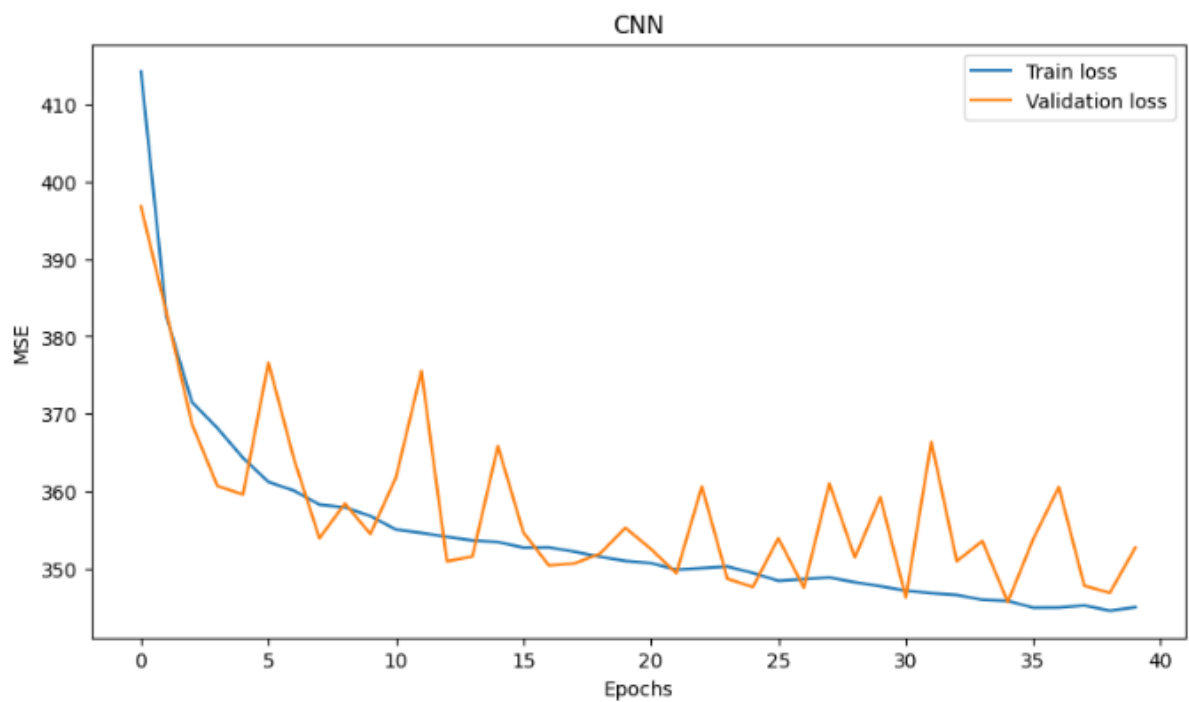


Figure 12-RMSE - CNN

- **CNN-LSTM** : meilleure précision dans les séquences complexes, mais parfois trop ajusté à l'entraînement.

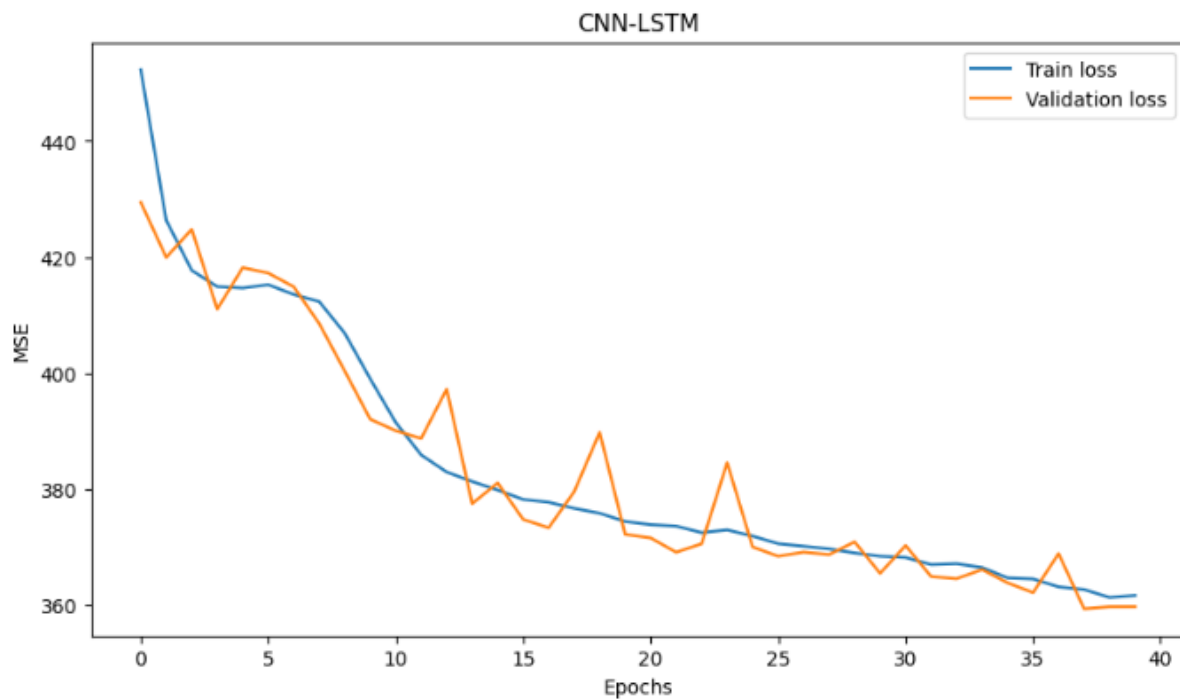


Figure 13-RMSE –(CNN-LSTM)

6. Synthèse Comparative :

Modèle	Simplicité	Nombre de Paramètres	RMSE (validation)	Interprétation
MLP	✓✓✓	Faible (3,201)	18.67164	Très bon rapport complexité/performance
CNN	✓✓	Moyen (45,093)	18.77945	Bon pour motifs locaux
CNN-LSTM	✓	Élevé (99,979)	18.96548	Puissant mais légèrement surappris

Tableau 5 - Synthèse comparative

7. Rôle du Chatbot IA dans l'Analyse :

Le **chatbot intelligent intégré** permet à l'utilisateur d'interroger dynamiquement les résultats. Il peut :

- Expliquer les différences entre les modèles.
- Aider à interpréter les erreurs.
- Recommander un modèle selon le contexte d'utilisation.
- Fournir des résumés personnalisés des résultats.

Cela améliore l'expérience utilisateur et démocratise l'analyse prédictive.

8. Conclusion de partie III :

Les résultats montrent que **le modèle MLP** offre les **meilleures performances** pour ce jeu de données, en raison de sa **simplicité et de sa capacité de généralisation**. Cependant, les modèles CNN et CNN-LSTM peuvent devenir plus avantageux pour des jeux de données plus volumineux ou plus complexes. Le choix du modèle doit donc être guidé par les besoins métier, les ressources disponibles et les caractéristiques des données.

Conclusion Générale et Perspectives :

Conclusion Générale :

Dans un contexte où la compétitivité des entreprises repose de plus en plus sur leur capacité à exploiter intelligemment les données commerciales, ce mémoire a proposé la conception et la mise en œuvre d'une **plateforme intelligente d'analyse et de prévision des ventes**, basée sur **Flask** et intégrant des **modèles d'intelligence artificielle avancés (MLP, CNN, CNN-LSTM)** ainsi qu'un **chatbot conversationnel IA** via l'API Groq.

L'architecture de la plateforme repose sur une structure modulaire, avec des interfaces conviviales pour le téléchargement de fichiers CSV, la visualisation des données, l'exécution de prédictions, et l'interaction avec un assistant virtuel capable de commenter les résultats en langage naturel. Ce système a été conçu pour être **accessible, évolutif et pertinent pour les décideurs**.

Sur le plan expérimental, trois modèles ont été comparés en termes de précision (via la RMSE) et de complexité :

- Le **MLP** s'est distingué par sa **simplicité et son efficacité**, affichant les meilleurs résultats de RMSE avec un très faible nombre de paramètres.
- Le **CNN** a montré une bonne capacité à extraire les motifs temporels récurrents, mais avec un léger surapprentissage.
- Le **CNN-LSTM**, plus complexe, a offert une modélisation riche mais au prix d'un ajustement excessif aux données d'entraînement.

Cette analyse comparative met en lumière l'importance de choisir un modèle adapté non seulement aux données mais aussi aux ressources disponibles et aux objectifs métiers. Le système développé démontre ainsi qu'il est possible de construire des outils prédictifs puissants, interprétables et intégrés dans une interface web moderne.

Enfin, l'ajout d'un **chatbot IA** représente une avancée significative en matière d'**accessibilité des analyses avancées**, en permettant aux utilisateurs non techniques de bénéficier d'explications, de conseils et de recommandations personnalisées sur les données et les résultats.

Perspectives :

Ce travail, bien qu'abouti, ouvre la voie à plusieurs pistes d'amélioration et d'approfondissement :

1. Amélioration des performances prédictives :

L'intégration de techniques avancées telles que **l'optimisation bayésienne**, **les Transformers temporels** ou **les modèles hybrides à attention** pourrait améliorer la précision et la robustesse des prédictions.

2. Généralisation multi-domaine :

La plateforme pourrait être adaptée à d'autres types de données temporelles : **flux logistiques**, **taux d'occupation**, **trafic web**, etc., en ajustant les hyperparamètres et la structure des modèles.

3. Personnalisation du chatbot :

Le chatbot pourrait être renforcé par l'apprentissage continu sur les interactions passées ou par l'ajout de **modules de recommandation**, **d'analyse causale**, ou **de génération de rapports automatiques**.

4. Déploiement en production :

Une future version pourrait être **hébergée dans le cloud**, avec **authentification sécurisée**, **multi-utilisateurs**, et **stockage en base de données**, facilitant ainsi son utilisation dans des environnements professionnels réels.

5. Interprétabilité des modèles :

L'intégration de techniques comme **SHAP** ou **LIME** permettrait d'expliquer plus finement les décisions des modèles, renforçant la **confiance** des utilisateurs dans les prédictions.

Mot de la fin :

En synthèse, ce projet incarne l'union de l'intelligence artificielle, de l'ingénierie logicielle et de l'expérience utilisateur pour répondre à un besoin concret : **mieux comprendre et anticiper les ventes**. La démarche adoptée, alliant rigueur scientifique et pragmatisme applicatif, offre une base solide pour développer des outils d'aide à la décision intelligents, accessibles et performants, dans un monde où les données sont plus que jamais un levier stratégique.

Bibliographie :

- Abadi, M., Barham, P., Chen, J., et al. (2016). *TensorFlow: A system for large-scale machine learning*. In *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, 265–283.
- Borovykh, A., Bohte, S., & Oosterlee, C. W. (2017). *Conditional time series forecasting with convolutional neural networks*. arXiv preprint arXiv:1703.04691.
- Bontempi, G., Taieb, S. B., & Le Borgne, Y. A. (2013). *Machine learning strategies for time series forecasting*. In *European Business Intelligence Summer School* (pp. 62–77). Springer.
- Brown, T., Mann, B., Ryder, N., et al. (2020). *Language models are few-shot learners*. *Advances in Neural Information Processing Systems*, 33, 1877–1901.
- Brownlee, J. (2017). *Deep Learning for Time Series Forecasting: Predict the Future with MLPs, CNNs and LSTMs in Python*. Machine Learning Mastery.
- Cao, Y., Wang, L., Li, X., & He, J. (2020). *Hybrid deep learning model for time series forecasting: Application to short-term load forecasting*. *IEEE Access*, 8, 163679–163693.
- Choi, H., & Varian, H. (2012). *Predicting the present with Google Trends*. *Economic Record*, 88(s1), 2–9.
- Chollet, F. (2017). *Deep Learning with Python*. Manning Publications.
- Fischer, T., & Krauss, C. (2018). *Deep learning with long short-term memory networks for financial market predictions*. *European Journal of Operational Research*, 270(2), 654–669.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
- Groq. (2024). *Groq API documentation*. <https://groq.com/docs>
- Hochreiter, S., & Schmidhuber, J. (1997). *Long short-term memory*. *Neural Computation*, 9(8), 1735–1780.
- Hunter, J. D. (2007). *Matplotlib: A 2D graphics environment*. *Computing in Science & Engineering*, 9(3), 90–95.
- Kaggle. (n.d.). <https://www.kaggle.com>
- LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). *Gradient-based learning applied to document recognition*. *Proceedings of the IEEE*, 86(11), 2278–2324.
- Microsoft. (2022). *Flask Web Framework*. <https://flask.palletsprojects.com>
- OpenAI. (2023). *LLaMA Models: Language Modeling with Transformers*. <https://openai.com/research>
- Pedregosa, F., Varoquaux, G., Gramfort, A., et al. (2011). *Scikit-learn: Machine learning in Python*. *Journal of Machine Learning Research*, 12, 2825–2830.
- Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). *Learning representations by back-propagating errors*. *Nature*, 323(6088), 533–536.
- Russell, S., & Norvig, P. (2016). *Artificial Intelligence: A Modern Approach* (3rd ed.). Pearson Education.

- Zhang, G., Eddy Patuwo, B., & Hu, M. Y. (1998). *Forecasting with artificial neural networks: The state of the art*. *International Journal of Forecasting*, 14(1), 35–62.
- Zhao, H., Yuan, D., & Lin, C. (2017). *Sales forecasting using convolutional neural network*. In *2017 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, 1235–1239.

Annexe :

Annexe A – Extrait du jeu de données utilisé :

Le tableau ci-dessous représente un extrait du jeu de données *sample_sales.csv* utilisé dans notre projet. Ce fichier contient les ventes quotidiennes par produit et par magasin. Il a servi à alimenter les modèles MLP, CNN et CNN-LSTM dans l'étape de prévision :

Date	Item	Store	Sales
2017-01-01	1	1	20
2017-01-02	1	1	22
2017-01-03	1	1	19
2017-01-04	1	1	21
2017-01-05	1	1	18
2017-01-06	1	1	23
2017-01-07	1	1	24
2017-01-08	1	1	22
2017-01-09	1	1	20
2017-01-10	1	1	19

Tableau 6 - jeu de données

Description des colonnes :

- **Date** : la date de la vente (au format AAAA-MM-JJ).
- **Item** : identifiant unique du produit vendu.
- **Store** : identifiant du magasin où la vente a eu lieu.
- **Sales** : nombre d'unités vendues ce jour-là.

Annexe B – Organisation des fichiers de l'application :

Voici l'architecture simplifiée de l'application Flask développée :

Fichier / Dossier	Description
app.py	Fichier principal contenant toute la logique applicative
templates/	Contient les fichiers HTML (pages web)
static/	Fichiers statiques (CSS, JS, images)
uploads/	Stockage temporaire des fichiers CSV chargés
model_cnn_lstm.h5	Modèle pré-entraîné basé sur l'architecture CNN-LSTM
model_mlp.h5	Modèle MLP sauvegardé après entraînement
model_cnn.h5	Modèle CNN sauvegardé

Tableau 7 - Organisation des fichiers de l'application

Annexe C – Environnement de développement :

Cette section présente les outils, bibliothèques et langages utilisés dans le développement de la plateforme d'analyse prédictive des ventes.

Élément	Détails techniques
Langage de programmation	Python 3.11
Environnement de développement	Visual Studio Code
Framework Web	Flask (micro-framework Python)
Bibliothèques de traitement	Pandas, NumPy, Matplotlib, Seaborn, Scikit-learn
Bibliothèques d'IA / Deep Learning	TensorFlow, Keras
Interface utilisateur	HTML5, CSS3, Jinja2
Visualisation interactive	Seaborn, Matplotlib
API IA pour chatbot	GROQ API (modèle LLaMA 3.3 - 70B)
Hébergement local	Flask, exécuté en mode debug pour tests locaux

Tableau 8 - Environnement de développement

Annexe D – Interfaces de la plateforme Flask :

Dans cette section, nous présentons les principales interfaces graphiques de la plateforme web développée avec **Flask**, destinée à l'analyse et à la prévision des ventes. Ces interfaces ont été pensées pour être intuitives, interactives et adaptées aux utilisateurs non techniques.

D.1. Interface d'accueil :

Cette page présente un aperçu global du projet, ses objectifs, et oriente l'utilisateur vers les différentes fonctionnalités de la plateforme.

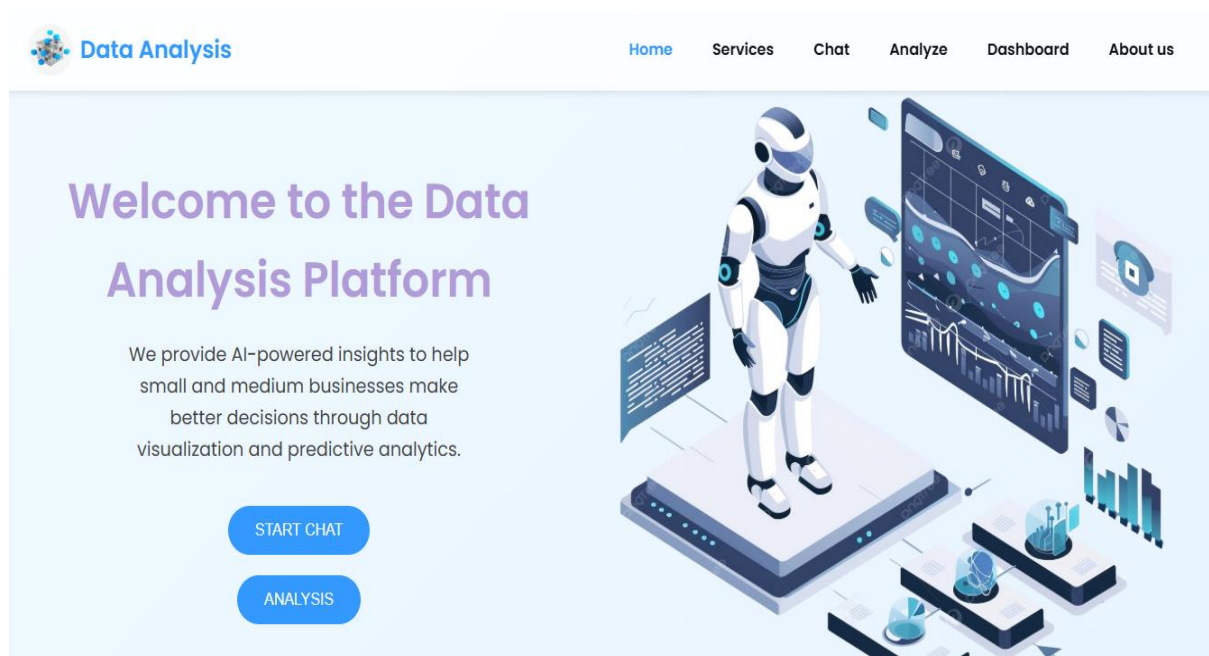


Figure 14-Interface d'accueil

D.2. Interface d'upload et de dashboard :

Permet à l'utilisateur de télécharger un fichier CSV contenant les données de ventes. Une fois le fichier envoyé, un tableau de bord dynamique affiche les visualisations suivantes :

- Histogrammes des variables numériques
- Diagrammes en boîte (boxplots)

- Graphiques en barres et camemberts pour les variables catégorielles
- Évolution temporelle des ventes

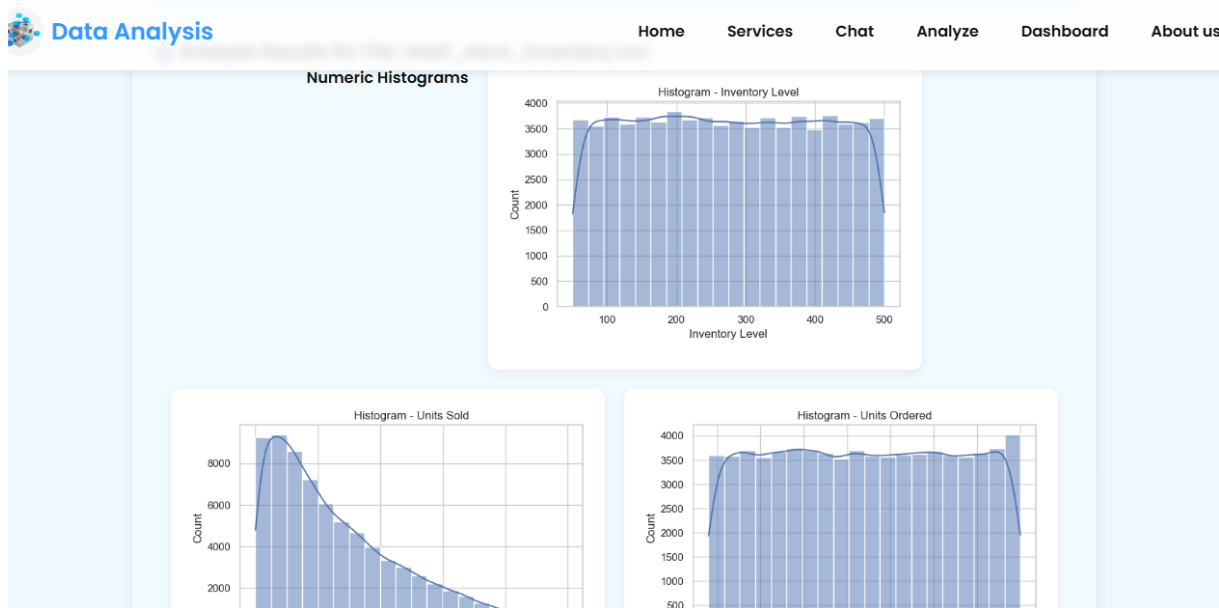


Figure 15-Interface d'upload et de dashboard

D.3. Interface de prédiction avec CNN-LSTM :

C'est l'interface principale pour la prévision des ventes. Voici les fonctionnalités clés :

- L'utilisateur sélectionne un fichier CSV contenant les colonnes date, item, store, sales.
- Le système prépare les données sous forme de séries temporelles.
- Le modèle **CNN-LSTM** est utilisé pour prédire l'évolution future des ventes.

Résultats affichés :

- Tableaux HTML des prévisions
- Graphiques montrant la comparaison entre les ventes réelles et les prédictions pour chaque article et chaque magasin
- Une explication succincte est fournie pour interpréter les résultats

Prediction Charts:

The chart shows a comparison between actual values and future predictions for each product in each store.

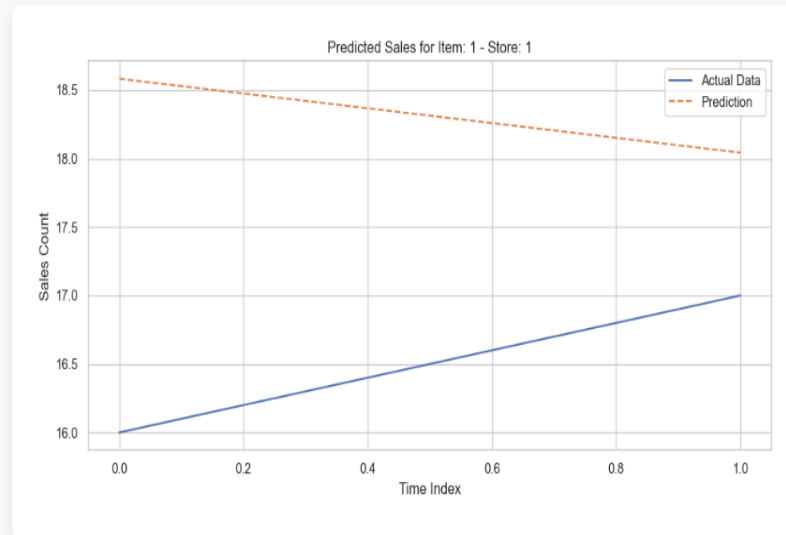


Figure 16-resultat de prediction CNN-LSTM

D.4. Interface du Chatbot IA :

Cette section permet à l'utilisateur de poser des questions sur ses données après téléchargement d'un fichier. Le **chatbot LLaMA 3** via l'API de **Groq** fournit des réponses précises basées exclusivement sur les données présentes.

- Le chatbot peut identifier les tendances, anomalies, colonnes manquantes, etc.
- Il répond en français ou en arabe selon la question.

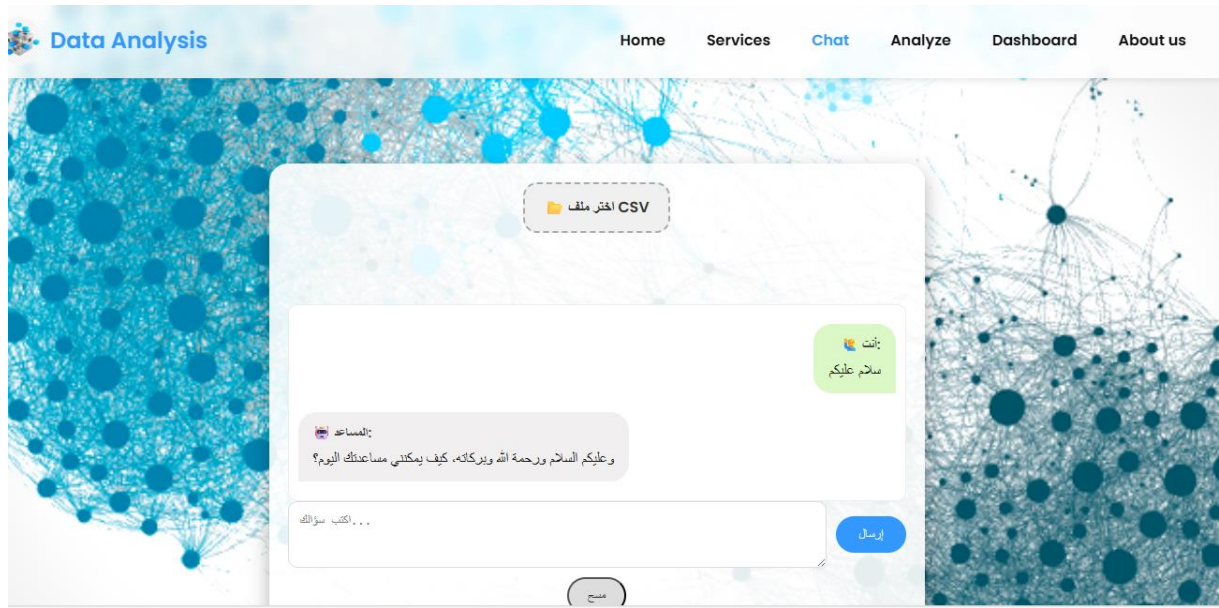


Figure 17-interface de chatbot IA

ملخص

يعرض هذا البحث تصميم وتطوير منصة ويب ذكية لتحليل المبيعات وتوقعها، تم تطويرها باستخدام إطار العمل Flask. تتيح المنصة للمستخدمين تحميل ملفات CSV، تحليل البيانات، وتوقع المبيعات المستقبلية باستخدام نماذج التعلم العميق (MLP)، CNN، LSTM، و (CNN-LSTM). تم تقييم أداء هذه النماذج باستخدام مقياس RMSE، وقد أظهر نموذج MLP أفضل أداء. كما تم دمج روبوت دردشة ذكي يعتمد على نموذج LLaMA 3 من خلال واجهة برمجة التطبيقات Groq، لتقديم المساعدة في تفسير النتائج. تمثل هذه المنصة أداة شاملة تساعد الشركات على استغلال بياناتها التجارية بكفاءة.

الكلمات المفتاحية: تحليل المبيعات، التنبؤ، Flask، CNN، LSTM، MLP، CNN-LSTM، روبوت دردشة ذكي، API Groq

Abstract

This thesis presents the design of an intelligent web platform for sales analysis and forecasting, developed using the Flask framework. It enables users to upload CSV files, analyze data, and predict future sales using deep learning models (MLP, CNN, LSTM, and CNN-LSTM). The models' performance was evaluated using RMSE, with the MLP model showing the highest accuracy. Additionally, a chatbot based on LLaMA 3, integrated via the Groq API, assists users in interpreting the results. The platform serves as a comprehensive tool for businesses aiming to intelligently leverage their sales data.

Keywords: Sales analysis, Forecasting, Flask, CNN, LSTM, MLP, CNN-LSTM, AI Chatbot, Groq API.

Résumé

Ce mémoire décrit la conception d'une plateforme web intelligente d'analyse et de prévision des ventes, développée avec Flask. Elle permet de charger des fichiers CSV, d'analyser les données, et de prédire les ventes à l'aide de modèles de deep learning (MLP, CNN, LSTM et CNN-LSTM). Les performances des modèles ont été évaluées via le RMSE, montrant que le MLP est le plus précis. Un chatbot basé sur LLaMA 3 via l'API Groq est intégré pour assister les utilisateurs dans l'interprétation des résultats. La plateforme constitue ainsi un outil complet pour les entreprises souhaitant exploiter leurs données commerciales de manière intelligente.

Mots-clés : Analyse des ventes, Prévision, Flask, CNN, LSTM, MLP, CNN-LSTM, Chatbot IA, API Groq.