

République Algérienne Démocratique et Populaire
Ministère de l'enseignement supérieur et de la recherche scientifique
Université de 8 Mai 1945 – Guelma -
Faculté des Mathématiques, d'Informatique et des Sciences de la matière
Département d'Informatique



Mémoire de Fin d'études Master

Filière : Informatique

Option : Sciences et technologies de l'information et de la communication

Thème _____

Un système basé sur l'apprentissage automatique pour la prédiction de la
valeur des cryptomonnaies

Présenté Par :

OUNADI Fouad

Encadré par :

DR. CHOIRA Chemseddine

Membres de jury :

DR. BENHAMZA Karima

DR. BOURSSACE Hassina

Juin 2023

Remerciements

En entamant ces quelques lignes de remerciement, j'exprime ma gratitude à Dieu pour m'avoir accordé le courage, la persévérance et les ressources nécessaires pour mener à bien ce travail. Alhamdoulillah qui m'a facilité mon périple et qui m'a fait sortir des ténèbres de l'ignorance à la lumière de la science et de la connaissance. Alhamdoulillah que j'ai pu finalement arriver ici et que j'ai pu aboutir à un de mes objectifs dans la vie.

Je souhaite également exprimer ma profonde reconnaissance envers mes chers parents et mon frère. Leur amour, leur soutien indéfectible et leurs encouragements constants ont été des piliers essentiels tout au long de cette aventure académique. Je suis profondément reconnaissant d'avoir été soutenu par leur amour inconditionnel.

De plus, je tiens à remercier mon encadreur, Dr. CHOHRA Chemseddine, pour son précieux soutien, ses conseils éclairés et son engagement tout au long de mon travail. Son expertise, sa disponibilité et sa bienveillance ont joué un rôle crucial dans la réussite de ce projet. Je lui exprime ma gratitude pour sa patience et son accompagnement.

Je souhaite donc adresser mes chaleureux remerciements à l'ensemble des membres du jury pour leur contribution précieuse et leurs retours constructifs. Leurs évaluations ont grandement contribué à mon développement académique et professionnel.

Enfin, j'exprime ma gratitude envers tous mes amis, ma famille et mes proches qui m'ont soutenu tout au long de ce parcours académique. Leurs encouragements, leurs mots d'encouragement et leur présence bienveillante ont été une source de motivation essentielle lors des moments difficiles.

Je suis conscient que sans l'aide de chacune de ces personnes, cette réalisation n'aurait pas été possible. Leur soutien et leur confiance en moi ont été des éléments clés de mon succès.

RÉSUMÉ

Les cryptomonnaies ont en effet connu un intérêt et une popularité croissants au cours des dernières années en tant qu'actifs numériques indépendants des institutions bancaires centrales. Dans notre recherche et développement de notre modèle, nous avons principalement axé nos efforts sur la prédiction du prix du Bitcoin, la cryptomonnaie phare et la plus largement reconnue. Nous avons essayé plusieurs modèles entraînés sur différents datasets et indicateurs économiques et financiers, et les meilleurs résultats ont été obtenus en utilisant un modèle MLP (Multi-Layer Perceptron) entraîné sur une base de données contenant différentes variations de l'indicateur économique RSI (Relative Strength Index) calculées sur quatre périodes différentes.

Mots Clés : Cryptomonnaies, Bitcoin, Indicateurs, MLP, RSI

ABSTRACT

Cryptocurrencies have indeed experienced growing interest and popularity in recent years as digital assets independent of central banking institutions. In our research and development of our model, we have primarily focused our efforts on predicting the price of Bitcoin, the flagship and most widely recognized cryptocurrency. We have tried several models trained on different datasets and economic and financial indicators, and the best results were obtained using an MLP (Multi-Layer Perceptron) model trained on a database containing various variations of the RSI (Relative Strength Index) economic indicator calculated over four different periods.

Keywords : Cryptocurrency, Bitcoin, Indicators, MLP, RSI

TABLE DES MATIÈRES

Liste des figures		vii
Liste des tableaux		viii
1 APPRENTISSAGE AUTOMATIQUE		1
1.1 Introduction		1
1.2 Définition		1
1.3 Apprentissage supervisé		3
1.3.1 Classification		3
1.3.2 Régression		4
1.4 Apprentissage non supervisé		4
1.4.1 Regroupement (Clustering)		4
1.4.2 Association		5
1.5 Apprentissage par renforcement		5
1.6 Algorithmes d'apprentissage automatique		6
1.6.1 Logistique régression		6
1.6.2 Régression linéaire		6
1.6.3 Arbre de décision		7
1.6.4 K-Means		7

1.6.5	Machine à vecteur de support (SVM)	8
1.6.6	Naive Bayes	8
1.6.7	Random forest	9
1.6.8	K-Nearest Neighbour (KNN)	9
1.6.9	Analyse discriminante linéaire (LDA)	9
1.6.10	L'apprentissage profond	10
1.7	Réseaux de neurones artificiels (ANNs)	10
1.8	Types de réseaux de neurones	11
1.8.1	Perceptron Multi-Couches (MLP)	12
1.8.2	Convolutional Neural Network (CNN)	13
1.8.3	Réseau de neurones à fonctions de base radiales (RBF)	13
1.8.4	Réseau de neurones récurrents (RNN)	14
	Les réseaux neuronaux récurrents (LSTM)	15
	Réseau récurrent à portes (GRU)	18
1.9	Conclusion	20
2	Crypto-monnaies et prédiction de marché	21
2.1	Introduction	21
2.2	Définition	22
2.3	L'évolution des crypto-monnaies	22
2.4	Trading de crypto-monnaie	22
2.5	Avantages et inconvénients du trading des crypto-monnaies	23
	Les avantages	23
	Les inconvénients	24
2.6	Prédiction du marché	25
2.6.1	Indicateurs	25
2.6.2	Difficultés	27
2.7	Travaux connexes	28
2.8	Conclusion	32

3	Conception	33
3.1	Introduction	33
3.2	Architecture générale du système	33
3.2.1	La collection de l'ensemble de données	34
	OHLC (Open-High-Low-Close)	34
	Calcul du RSI	35
3.2.2	Prétraitement de données	36
3.2.3	Organiser et Diviser l'ensemble données	37
3.2.4	Phase d'apprentissage	38
3.2.5	Prédiction	38
3.3	Conclusion	39
4	Implémentation et résultats	41
4.1	Introduction	41
4.2	Implémentation	41
4.2.1	Environnement de développement	42
4.2.2	Langage de programmation et bibliothèques	42
	Python 3.9.12	42
	Bibliothèques utilisées	42
4.2.3	Implémentation de l'architecture proposé	43
	La collection de l'ensemble de données	43
	Prétraitement de données	44
	Diviser les données	44
	Phase d'apprentissage	44
4.2.4	Interface du système	45
4.3	Résultats et discussion	49
4.3.1	Métriques de performance	49
4.3.2	Résultats	51
4.4	Conclusion	54

TABLE DES FIGURES

1.1	Types d'apprentissage automatique [2].	2
1.2	Réseaux de neurones artificiels (ANNs) [19].	11
1.3	Réseau de neurones récurrents (RNN) [25].	14
1.4	Architecture de réseaux neuronaux récurrents avec mémoire à long terme (LSTM) [27].	16
1.5	Architecture de réseau récurrent à portes (GRU) [29].	19
3.1	Architecture générale du système	34
3.2	Organiser et diviser l'ensemble de données	37
3.3	Processus de prédiction	39
4.1	La fenêtre principale de l'application	46
4.2	Fenêtre de sélection des Dates	47
4.3	Résultats de la prédiction	48
4.4	Visualiser le graphique	48
4.5	Le Taux d'erreur (loss) de training et validation	51
4.6	Comparaison des résultats de prédiction à partir de différentes zones de départ	52
4.7	Situation imprédictible - Faillite de FTX	53

LISTE DES TABLEAUX

4.1	Le Taux d'erreur (loss) de training et validation	52
4.2	Résultats obtenus	54

INTRODUCTION GÉNÉRALE

Les cryptomonnaies ont suscité un grand intérêt ces dernières années, attirant les investisseurs, les passionnés de technologie et le grand public. Bitcoin en particulier s'est imposé comme un leader dominant et a acquis une renommée mondiale.

L'intérêt croissant pour les cryptomonnaies peut être attribué à plusieurs facteurs. Tout d'abord, elles offrent une alternative aux systèmes financiers traditionnels et aux devises contrôlées par les banques centrales. Les cryptomonnaies reposent sur la technologie de la blockchain, ce qui garantit des transactions sécurisées, transparentes et immuables, sans nécessiter d'intermédiaires.

De plus, les cryptomonnaies sont souvent considérées comme des investissements potentiellement rentables. Elles ont connu de grandes fluctuations de prix, attirant ainsi les commerçants et les investisseurs à la recherche de rendements élevés sur un marché volatil.

La prédiction précise des prix du Bitcoin est devenue un sujet d'intérêt et d'importance majeure dans les industries financières et d'investissement. Cependant, la forte volatilité et la sensibilité des cryptomonnaies à de nombreux facteurs en font un marché complexe à prévoir. Les investisseurs et les analystes sont confrontés au défi de développer des modèles fiables et des méthodes de prédiction pour prendre des décisions éclairées. Comment pouvons-nous améliorer les techniques de prédiction des prix des cryptomonnaies et surmonter les obstacles liés à la volatilité? Quelles sont les

variables et les indicateurs pertinents à prendre en compte pour obtenir des prédictions précises? Cette problématique nous pousse à explorer de nouvelles méthodes et à relever les défis liés à la prédiction des prix des cryptomonnaies.

Dans ce travail, nous explorons le domaine de la prédiction des prix de Bitcoin et proposons l'utilisation d'une approche puissante appelée modèle Perceptron Multi-Couche (MLP). Le modèle MLP offre des avantages distincts pour capturer les relations complexes et non linéaires, extraire des caractéristiques pertinentes et s'adapter à la dynamique en constante évolution du marché des cryptomonnaies. En incorporant l'un des indicateurs techniques couramment utilisés, nous visons à améliorer les capacités prédictives du modèle. Cet indicateur fournit des informations précieuses sur les conditions du marché, évalue la force et le momentum des mouvements de prix et aide à identifier les retournements de prix potentiels.

Grâce à cette recherche, nous souhaitons contribuer au domaine de la prédiction des cryptomonnaies et poser les bases d'avancées ultérieures pour prédire avec précision le prix de Bitcoin.

Nous divisons ce rapport en 4 chapitres. Dans le premier chapitre nous introduisons le concept de l'apprentissage automatique et les différents algorithmes qu'on peut utiliser pour implémenter un système de prédiction des prix des cryptomonnaies. Dans le deuxième chapitre nous expliquons la différence entre les cryptomonnaies et les monnaies traditionnelles et présentons quelques travaux connexes à notre sujet. Dans le chapitre 3 nous expliquons l'architecture de notre système en justifiant les choix que nous avons faits, et finalement dans le quatrième chapitre, nous présentons notre environnement de travail et les résultats que nous avons obtenus.

CHAPITRE 1

APPRENTISSAGE AUTOMATIQUE

1.1 Introduction

Pour faciliter l'accomplissement de diverses tâches, les humains ont utilisé un large éventail d'outils. L'inventivité du cerveau humain a conduit au développement de plusieurs inventions. En permettant aux humains de répondre à toute une série de besoins, notamment ceux liés aux déplacements, à l'industrie et à l'informatique, ces dispositifs ont amélioré la qualité de la vie humaine. En outre, l'apprentissage automatique fait partie de ces inventions. L'apprentissage automatique et l'apprentissage humain sont comparables. Et qui se base sur nécessite soit d'apprendre de nouvelles attitudes, principes, compétences ou préférences - soit de changer celles qui existent déjà. Le béhaviorisme, le cognitivisme, le constructivisme, l'expérialisme et l'apprentissage social sont des caractéristiques de la théorie de l'apprentissage personnel.

1.2 Définition

L'apprentissage automatique (ou Machine Learning - ML) est un sous-ensemble de l'intelligence artificielle qui donne aux ordinateurs la capacité de raisonner et d'apprendre par eux-mêmes. L'objectif est de faire en sorte que les ordinateurs modifient

leur comportement afin d'améliorer ce dernier et d'en accroître l'exactitude. Selon Arthur Samuel, le domaine d'étude qui permet aux ordinateurs d'apprendre sans être explicitement enseignés est connu sous le nom d'apprentissage automatique. Récemment, Tom Mitchell a fourni un concept "bien posé" qui a été plus utile à la configuration de l'ingénierie : "On dit d'un programme informatique qu'il apprend de l'expérience E par rapport à une certaine tâche T et une certaine métrique de performance". P , s'il obtient de meilleures performances sur T , telles que jugées par P , au fur et à mesure qu'il acquiert de l'expérience E [1]. Les algorithmes ML sont largement classés en trois catégories, à savoir l'apprentissage supervisé, l'apprentissage non supervisé et l'apprentissage par renforcement. montré dans la figure 3.1.

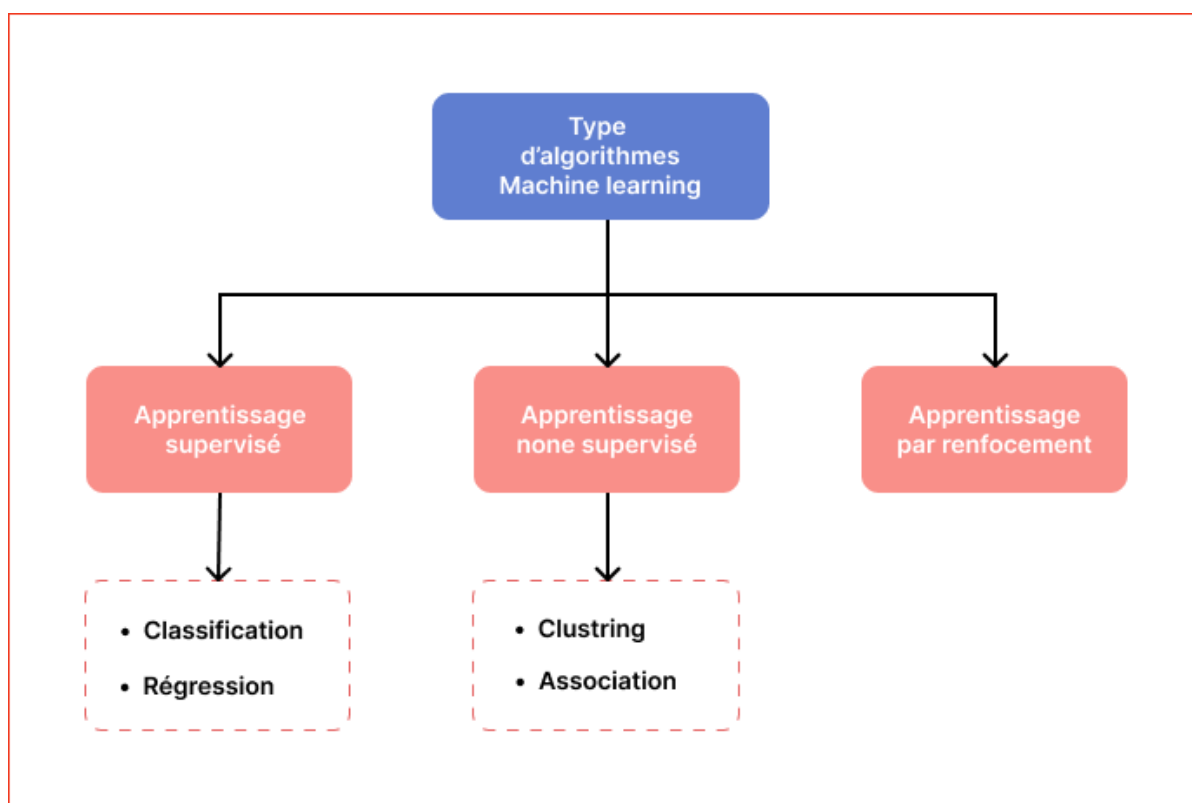


FIGURE 1.1 – Types d'apprentissage automatique [2].

1.3 Apprentissage supervisé

Les applications de l'apprentissage supervisé dans le monde réel ont été couronnées de succès. Il existe des similitudes entre ce type d'apprentissage et l'apprentissage humain. Pour accroître notre capacité à mener des activités dans le monde réel, nous pouvons apprendre de nos expériences passées. L'apprentissage automatique, quant à lui, apprend à partir de données qui ont été collectées dans le au passé et qui, dans certaines applications du monde réel représentent des expériences antérieures. brièvement, L'apprentissage supervisé implique l'apprentissage d'une correspondance entre un ensemble de variables d'entrée X et une variable de sortie Y et d'appliquer cette correspondance pour prédire les sorties pour des données non vues [3]. L'apprentissage supervisé est classé en deux catégories d'algorithmes :

1.3.1 Classification

En apprentissage automatique et en statistique, la classification est la tâche qui consiste à déterminer à quelle sous-population (groupe de catégories) appartient une nouvelle observation, étant donné l'ensemble d'apprentissage de données (ou instances) dont l'appartenance à une catégorie spécifique est connue. L'attribution d'un diagnostic à un patient et la classification d'un courriel comme "spam" ou "non-spam" sur la base de son contenu et de ses attributs en sont deux exemples (sexe, tension artérielle, présence ou absence de certains symptômes, etc.) La classification est considérée comme une illustration de l'apprentissage supervisé et de la reconnaissance des formes, c'est-à-dire un apprentissage dans lequel un ensemble de formations d'observations correctement identifiées est accessible. De même, le clustering est une méthode non supervisée. Une mesure de similarité ou de distance inhérente est utilisée pour catégoriser les données [4].

1.3.2 Régression

L'objectif des approches supervisées basées sur la régression est de modéliser clairement la relation entre les entrées ou les variables indépendantes et les sorties. Généralement, ces méthodes prennent la forme d'équations paramétriques avec des paramètres estimés à partir des données. Ces techniques fournissent fréquemment des estimations explicites des mesures d'association entre les entrées individuelles et le résultat, ajustées pour d'autres entrées, avec des estimations d'erreur standard fournies par le paradigme de modélisation utilisé [5].

1.4 Apprentissage non supervisé

Les algorithmes permettant de trouver des modèles dans des ensembles de données comprenant des points de données qui ne sont ni catégorisés ni étiquetés sont connus sous le nom d'apprentissage non supervisé. Ainsi, les algorithmes sont libres de catégoriser, d'étiqueter et d'organiser les points de données dans les ensembles de données sans avoir besoin d'une aide extérieure. Les utilisateurs n'ont pas besoin de surveiller le modèle. En d'autres termes, aucune étiquette n'est donnée à l'algorithme d'apprentissage; il est laissé à lui-même pour identifier la structure de la collection d'entrée. Même si aucune catégorie n'est fournie dans l'apprentissage non supervisé, un système d'IA regroupera le matériel non trié en fonction des similitudes et des différences. L'objectif principal de l'apprentissage non supervisé est de trouver des modèles cachés et intrigants dans des données non étiquetées [6].

1.4.1 Regroupement (Clustering)

Le clustering est une sorte d'apprentissage non supervisé basé sur la théorie de la similarité, qui regroupe les données très similaires dans les mêmes clusters et regroupe les données moins similaires dans des clusters différents. Contrairement à la

classification, les algorithmes de clustering ne nécessitent pas de connaissances préalables ou de données étiquetées [7].

1.4.2 Association

Comme les données ne sont pas étiquetées, les règles d'association utilisent l'apprentissage non supervisé pour tenter d'apprendre sans professeur. Les règles d'association sont descriptives et non prédictives, et sont généralement employées pour trouver des associations intrigantes enfouies dans d'énormes ensembles de données. Les règles ou les ensembles d'éléments communs sont fréquemment utilisés pour représenter les relations. L'extraction de règles d'association est utilisée pour trouver des connexions nouvelles et fascinantes entre divers éléments d'un ensemble, des modèles communs dans les données transactionnelles ou toute base de données relationnelle.

1.5 Apprentissage par renforcement

Dans un système d'apprentissage par renforcement, nous décrivons l'état actuel du système, spécifions un objectif, fournissons une liste d'actions autorisées et leurs contraintes environnementales pour leurs résultats, puis laissons le modèle ML expérimenter le processus de réalisation de l'objectif par lui-même en utilisant le principe d'essai et d'erreur pour maximiser une récompense. Au lieu de fournir des paires d'entrée et de sortie. Cependant, ils sont tout aussi pertinents pour les systèmes multi-agents comme les marchés électroniques. Les modèles d'apprentissage par renforcement ont été utilisés avec un succès remarquable dans des environnements fermés comme les jeux [8].

1.6 Algorithmes d'apprentissage automatique

De nombreux algorithmes d'apprentissage automatique différents sont utilisés par les scientifiques des données pour trouver des modèles dans les quantités massives de données qui mènent à des connaissances utiles. À un haut niveau, ces différents algorithmes peuvent être divisés en deux groupes en fonction de la façon dont ils "apprennent" les données pour faire des prédictions : l'apprentissage supervisé et l'apprentissage non supervisé. Nous citons ci-dessous certains des algorithmes d'apprentissage les plus utilisés.

1.6.1 Logistique régression

La régression logistique est essentiellement un algorithme de classification supervisée. Dans un problème de classification, la variable cible (ou sortie), y , ne peut prendre que des valeurs discrètes pour un ensemble donné de caractéristiques (ou entrées), X . Il s'agit d'une méthode fiable pour résoudre les problèmes de classification binaire. Afin d'estimer la probabilité qu'un résultat ait seulement deux valeurs, on utilise la régression logistique. La fonction logistique, une courbe en forme de S qui associe tout nombre à valeur réelle à une valeur comprise entre 0 et 1, mais jamais précisément à 0 et 1, est le fondement de la régression logistique [9].

1.6.2 Régression linéaire

La régression linéaire est un algorithme d'apprentissage automatique basé sur l'apprentissage supervisé, Il effectue une tâche de régression. Les modèles linéaires de classification séparent les vecteurs d'entrée en classes à l'aide de frontières de décision linéaires (hyperplans). Le but de la classification dans les classificateurs linéaires utilisés dans l'apprentissage automatique est de regrouper les éléments ayant des valeurs de caractéristiques comparables dans des groupes. Un classificateur linéaire atteint cet objectif en basant son choix de classification sur la valeur de la combinaison linéaire des caractéristiques. Étant donné qu'il est considéré comme le classificateur

le plus rapide, un classificateur linéaire est fréquemment employé dans des circonstances où la vitesse de classification est une préoccupation. De plus, les classificateurs linéaires donnent souvent d'excellents résultats lorsqu'il y a de nombreuses dimensions, comme dans la catégorisation de documents, où chaque dimension est généralement le nombre de comptes d'un mot dans un texte. Cependant, la marge affecte la vitesse de convergence des variables de l'ensemble de données. En général, la marge mesure la capacité d'un ensemble de données à être séparé linéairement, et donc la facilité avec laquelle il est possible de résoudre un problème de classification donnée [10].

1.6.3 Arbre de décision

L'apprentissage par arbre de décision est une notion qui explique une méthode permettant de créer des arbres de décision à partir de données afin d'effectuer des tâches de classification et de régression. Un arbre de décision est une structure arborescente qui ressemble à un organigramme, où chaque nœud interne représente un test sur un attribut, chaque branche représente un résultat de test, et chaque nœud feuille (nœud terminal) comprend une étiquette de classe [11].

1.6.4 K-Means

Technique de clustering, l'algorithme K-Means divise la collection de données non étiquetées en différents groupes. Ici, K spécifie combien de clusters doivent être formés au cours de la procédure ; par exemple, si $K = 2$, il y aura deux clusters, et pour Si $K = 3$, il y'en aura trois, et ainsi de suite. Elle nous donne la possibilité de classer les données en divers groupes et fournit une méthode pratique pour comprendre les catégories et les groupes dans l'ensemble de données non étiquetées sans avoir besoin d'entraînement. Chaque groupe est lié à un centroïde dans cette approche. L'objectif principal de cette technique est de réduire la distance totale entre chaque point de données et chacun de ses groupes pertinents. Le programme utilise l'ensemble de

données sans étiquettes d'entrée et divise l'ensemble de données en un nombre k de clusters, et répète le processus jusqu'à ce qu'il ne trouve plus de meilleure décomposition. La valeur de k doit être prédéterminée dans cet algorithme [12].

1.6.5 Machine à vecteur de support (SVM)

Les machines à vecteurs de support constituent une technique d'apprentissage automatique bien connue pour la classification et d'autres tâches d'apprentissage. Formellement, le SVM est un classificateur discriminant qui possède un hyperplan idéal. Les vecteurs de support sont ce qui est produit comme résultat de l'hyperplan idéal, qui classifie les nouveaux exemples et ensembles de données qui supportent l'hyperplan. Cet hyperplan est une ligne qui se divise en deux segments, chacun se trouvant de part et d'autre, dans la région bidimensionnelle (2D). Cependant, choisir le meilleur hyperplan est une tâche difficile car il ne doit pas être sensible au bruit et la généralisation des ensembles de données doit être précise. De manière pertinente, le SVM tente d'identifier un hyperplan optimal qui offre une distance minimale significative par rapport à l'ensemble de données d'apprentissage [13].

1.6.6 Naive Bayes

Les classificateurs bayésiens sont des classificateurs statistiques. Ils peuvent prévoir les probabilités d'appartenance à une classe, comme la probabilité qu'un échantillon donné appartienne à une classe spécifique. Le théorème de Bayes est le fondement du classificateur bayésien. Le principe sous-jacent des classifieurs bayésiens est que l'impact d'un attribut sur une classe particulière est indépendant des valeurs des autres attributs. L'indépendance conditionnelle de classe est le terme utilisé pour décrire cette présomption. Elle est considérée comme "naïve" dans ce sens car elle est conçue pour simplifier le calcul impliqué [14].

1.6.7 Random forest

Cette technique d'apprentissage d'ensemble permet de réaliser à la fois la classification et la régression. Elle crée une tonne d'arbres de décision à l'aide d'une méthodologie de mise en sac en utilisant une sélection aléatoire de données. Les arbres de décision qui sont finalement produits sont les résultats de la combinaison de tous les arbres de décision de la forêt aléatoire. L'algorithme de la forêt aléatoire se compose de deux étapes : la première crée la forêt aléatoire, et la seconde utilise le classificateur produit lors de la première étape pour établir des prévisions à l'aide de la forêt aléatoire [15].

1.6.8 K-Nearest Neighbour (KNN)

est l'une des techniques d'apprentissage par instance, qui appartient au groupe de l'apprentissage supervisé. Lorsqu'une nouvelle requête ou instance est lancée, un ensemble d'instances apparentées ou voisines est récupéré en mémoire et utilisé pour catégoriser la nouvelle instance. KNN ne fait que stocker les données d'entraînement fournies. Il est souvent utile de considérer de nombreux voisins lors de la classification, et cette méthode est connue sous le nom de k-voisin le plus proche. La distance entre les instances représentées par des entrées vectorielles, ainsi que plusieurs autres mesures connexes, sont utilisées pour déterminer quelles instances sont les plus proches les unes des autres. Cependant, la raison d'être de l'utilisation d'une distance pour catégoriser une nouvelle requête est que l'on s'attend à ce que les instances similaires aient une faible distance entre elles [16].

1.6.9 Analyse discriminante linéaire (LDA)

Pour les applications de classification supervisée, l'analyse discriminante linéaire est une méthode populaire de réduction de la dimensionnalité. Elle est utilisée pour modéliser les distinctions entre les groupements, c'est-à-dire pour faire la distinction

entre deux ou plusieurs classes. Elle est utilisée pour projeter les propriétés d'un espace de dimension supérieure dans un espace de dimension inférieure. Par exemple, nous avons deux classes et nous devons les diviser efficacement. Les classes peuvent avoir une variété d'attributs. Il peut y avoir un certain chevauchement lorsqu'on les classe sur la base d'un seul attribut. Par conséquent, nous continuerons à ajouter des qualités qui rendent une classification précise [17].

1.6.10 L'apprentissage profond

L'apprentissage profond, un sous-domaine de l'apprentissage automatique, utilise des réseaux neuronaux artificiels pour apprendre à partir de données. Ces réseaux neuronaux sont souvent composés de plusieurs couches de traitement, y compris des couches d'entrée, cachées et de sortie. La précision des modèles d'apprentissage profond est fortement influencée par le volume de données qui leur est fournie, et ils sont particulièrement efficaces lorsqu'ils travaillent avec de grands ensembles de données. Par rapport aux techniques traditionnelles d'apprentissage automatique, l'apprentissage profond est généralement plus performant, en particulier lorsqu'il s'agit de traiter des quantités importantes de données. Toutefois, l'efficacité de l'apprentissage profond peut varier en fonction des caractéristiques des données et de la configuration expérimentale [18].

1.7 Réseaux de neurones artificiels (ANNs)

Les réseaux neuronaux artificiels (ANN) sont des systèmes de traitement informatique qui tirent une grande partie de leurs fonctionnalités des systèmes nerveux organiques, comme le cerveau humain. Les ANNs sont principalement constitués d'un grand nombre de nœuds de calcul interconnectés (parfois appelés neurones), qui collaborent pour apprendre de l'entrée afin d'optimiser la sortie finale. La figure 1 représente un modèle de la structure fondamentale d'un ANN. L'entrée, qui est généralement chargée comme un vecteur multidimensionnel, est distribuée aux couches

cachées par la couche d'entrée. Ensuite, les couches cachées prennent en compte les décisions prises par la couche supérieure et déterminent si un changement stochastique en leur sein est bénéfique ou préjudiciable à la sortie finale, ce que l'on appelle le processus d'apprentissage. Le fait d'avoir plusieurs couches cachées empilées les unes sur les autres est communément appelé apprentissage profond [19].

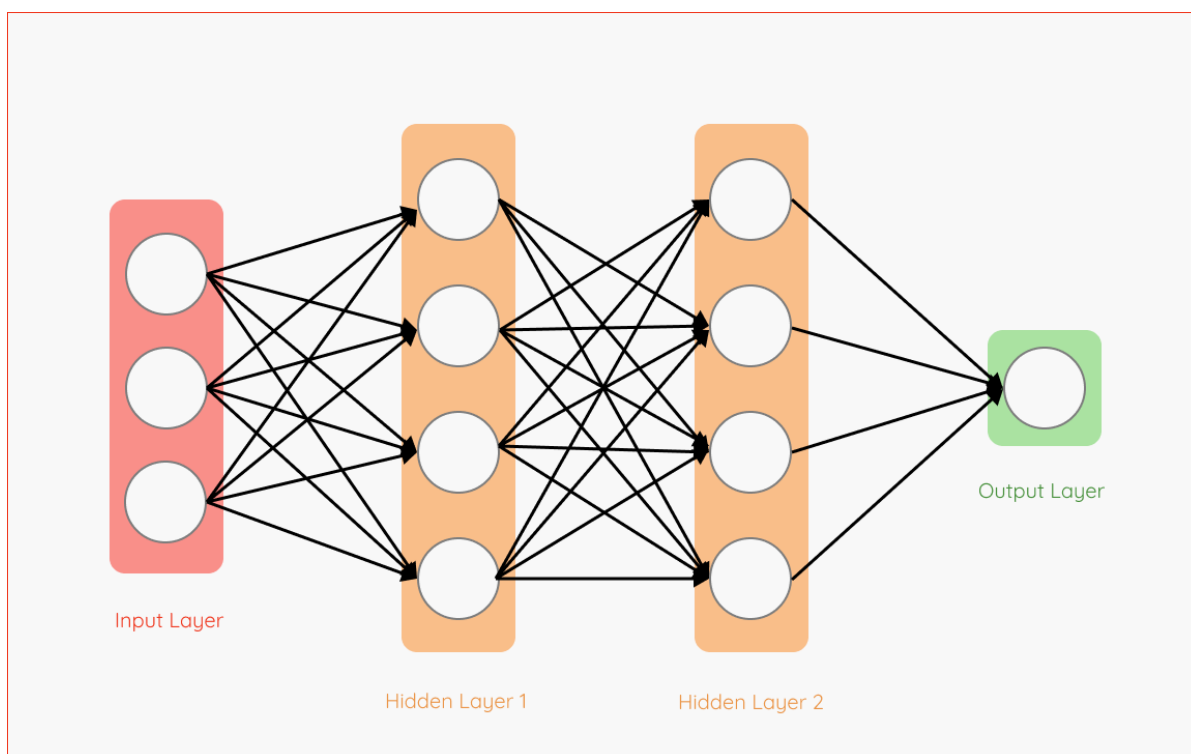


FIGURE 1.2 – Réseaux de neurones artificiels (ANNs) [19].

1.8 Types de réseaux de neurones

Les différentes formes de réseaux neuronaux sont des idées qui précisent comment l'architecture d'un réseau neuronal fonctionne dans un calcul, à l'image de la fonctionnalité du cerveau humain dans la prise de décisions. Les différents types de réseaux neuronaux comprennent :

1.8.1 Perceptron Multi-Couches (MLP)

Le MLP est un type d'ANN feedforward qui se compose d'une couche d'entrée, d'une ou plusieurs couches cachées et d'une couche de sortie. Les fonctions d'activation sont utilisées pour représenter la correspondance non linéaire complexe entre un ensemble de variables d'entrée et de sortie. Les poids et les fonctions de transfert/activation non linéaires relient complètement les nœuds des niveaux précédents et suivants. La mise à l'échelle de la sortie des nœuds de la couche précédente et le transfert vers l'avant permettent d'obtenir l'entrée d'un nœud de la couche suivante. Le MLP est plus apte à apprendre et à exprimer des relations non linéaires en raison de la superposition de fonctions de transfert non linéaires. Par conséquent, le plus grand avantage du MLP est sa rapidité à résoudre des problèmes complexes. Le MLP est une généralisation du perceptron qui surmonte la faiblesse de son incapacité à reconnaître des données linéaires et inséparables. [20].

Ces réseaux sont essentiellement une combinaison de neurones, de biais attribués aux neurones, de connexions ou liens entre eux et de poids attribués à ces connexions. Le processus d'apprentissage est effectué en fonction des ensembles de données d'entrée et de cibles, et à l'aide d'algorithmes d'entraînement.

Mathématiquement, un neurone K peut être défini par les deux équations suivantes :

$$y_k = f(u_k + b_k) \quad (1.1)$$

$$u_k = \sum_{i=1}^N w_{ki} x_i \quad (1.2)$$

Les signaux d'entrée $x_1, x_2, x_3, \dots, x_n$ représentent les signaux d'entrée, $w_{k1}, w_{k2}, w_{k3}, \dots, w_{kn}$ sont les poids de connexion du neurone, u_k est la sortie linéaire de la combinaison linéaire des entrées pondérées, b_k est le terme de biais, f est la fonction d'activation, et y_k est le signal de sortie du neurone.

Les réseaux de neurones à propagation avant (MLP-NNs) sont entraînés sur la base de l'algorithme de rétropropagation (BP) qui suit une procédure d'apprentissage basée sur la règle de correction d'erreur. En fait, le réseau produit des sorties de réseau en traitant les données d'entrée reçues. Ensuite, en comparant les valeurs cibles et la sortie du réseau, la valeur d'erreur est calculée. Ensuite, les poids et les biais sont ajustés pour minimiser l'erreur ; le processus d'entraînement se poursuit jusqu'à ce que le réseau atteigne une erreur minimale prédéfinie autorisée [21].

1.8.2 Convolutional Neural Network (CNN)

Les réseaux neuronaux convolutifs (CNN) sont des systèmes d'intelligence artificielle basés sur des réseaux neuronaux multicouches qui apprennent des informations pertinentes à partir d'images et sont capables d'effectuer une variété de tâches telles que la classification, la détection et la segmentation d'objets. L'avantage des CNN par rapport à d'autres algorithmes de classification (SVM, K-NN, Random-Forest et autres) est que les CNN apprennent les meilleures caractéristiques pour représenter les objets dans les images et ont une grande capacité de généralisation, étant capables de classer avec précision de nouveaux exemples avec seulement quelques exemples dans l'ensemble d'apprentissage [22].

1.8.3 Réseau de neurones à fonctions de base radiales (RBF)

Les fonctions de base radiales sont intégrées dans un réseau neuronal à deux couches de type feed-forward. Un réseau de ce type possède un ensemble particulier d'entrées et de sorties. Entre les entrées et les sorties se trouve une couche d'unités de traitement appelées unités cachées. Ils utilisent tous des fonctions de base radiales. Lors de l'estimation de séries chronologiques et de l'identification de modèles, le réseau est appliqué à la modélisation des données de nombreuses façons. la sortie unique du réseau représente une valeur de signal, tandis que les entrées du réseau

représentent des échantillons de données à des intervalles de temps historiques spécifiques. Les entrées d'une application de classification de modèles correspondent à des entrées de caractéristiques, et chaque sortie à une classe [23].

1.8.4 Réseau de neurones récurrents (RNN)

Le réseau neuronal artificiel récurrent est le nom donné à un réseau neuronal artificiel doté d'une topologie récurrente. Il est comparable à un réseau neuronal de type feed-forward, mais sans aucune restriction quant aux boucles arrière. Dans ce cas, l'information n'est plus seulement transférée dans un sens mais aussi en sens inverse. Par conséquent, le réseau développe un état interne qui lui permet d'afficher un comportement temporel dynamique. Les réseaux neuronaux artificiels récurrents peuvent traiter toute série d'entrées en utilisant leur mémoire interne [24].

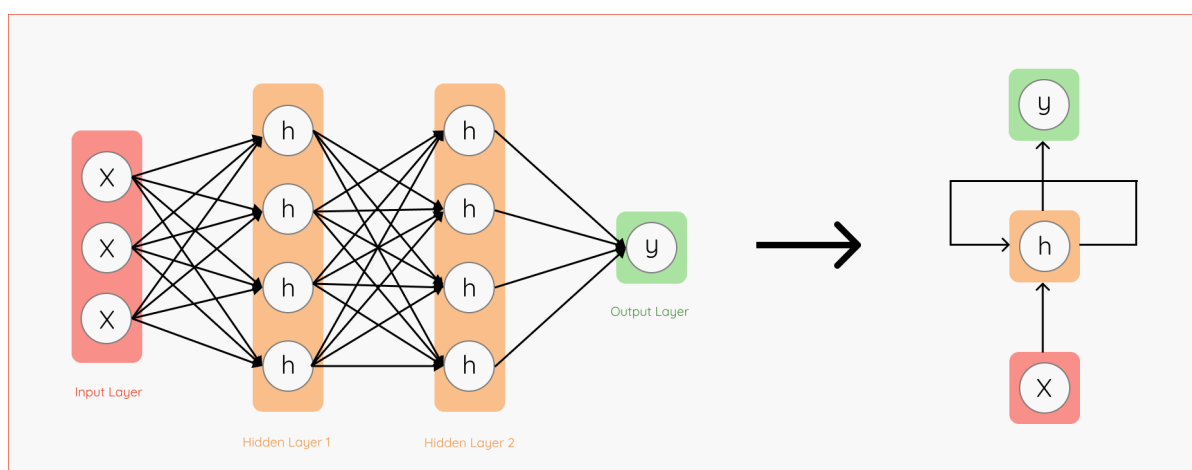


FIGURE 1.3 – Réseau de neurones récurrents (RNN) [25].

Il existe plusieurs types de RNN (réseaux neuronaux récurrents), qui diffèrent par leur architecture et leur utilisation. Voici quelques-uns des types de RNN les plus courants :

Les réseaux neuronaux récurrents (LSTM)

Les réseaux neuronaux récurrents avec mémoire à long terme (Long Short-Term Memory LSTM) se sont avérés être une solution efficace et évolutive pour une variété de problèmes d'apprentissage impliquant une entrée séquentielle. En raison de leur efficacité et de leur généralité, ils sont utiles pour capturer les dépendances temporelles à long terme. Une architecture inspirée des RNN, le LSTM possède des portes qui contrôlent le flux d'informations entre les cellules. La sortie finale étant une version filtrée de l'état de la cellule en fonction du contexte des entrées, les structures de portes d'entrée et d'oubli peuvent modifier l'information circulant dans l'état de la cellule [26]. Le LSTM est un type de réseau neuronal récurrent (RNN) qui a été spécialement conçu pour résoudre le problème de la disparition du gradient dans les RNN traditionnels. Ce problème se produit lors de la formation de RNN sur de longues séquences de données, où les gradients peuvent devenir trop petits pour être utiles à l'apprentissage. En utilisant une architecture plus complexe, le LSTM permet au réseau de se souvenir ou d'oublier sélectivement des informations à chaque pas de temps, ce qui permet de préserver des informations importantes sur des périodes plus longues. Le LSTM est ainsi plus efficace que les RNN traditionnels pour modéliser les données séquentielles, y compris les données de séries chronologiques telles que les prix des crypto-monnaies. Sa capacité à apprendre à partir de données séquentielles complexes en fait un choix populaire pour la prédiction des prix et l'analyse des tendances dans le marché des crypto-monnaies.

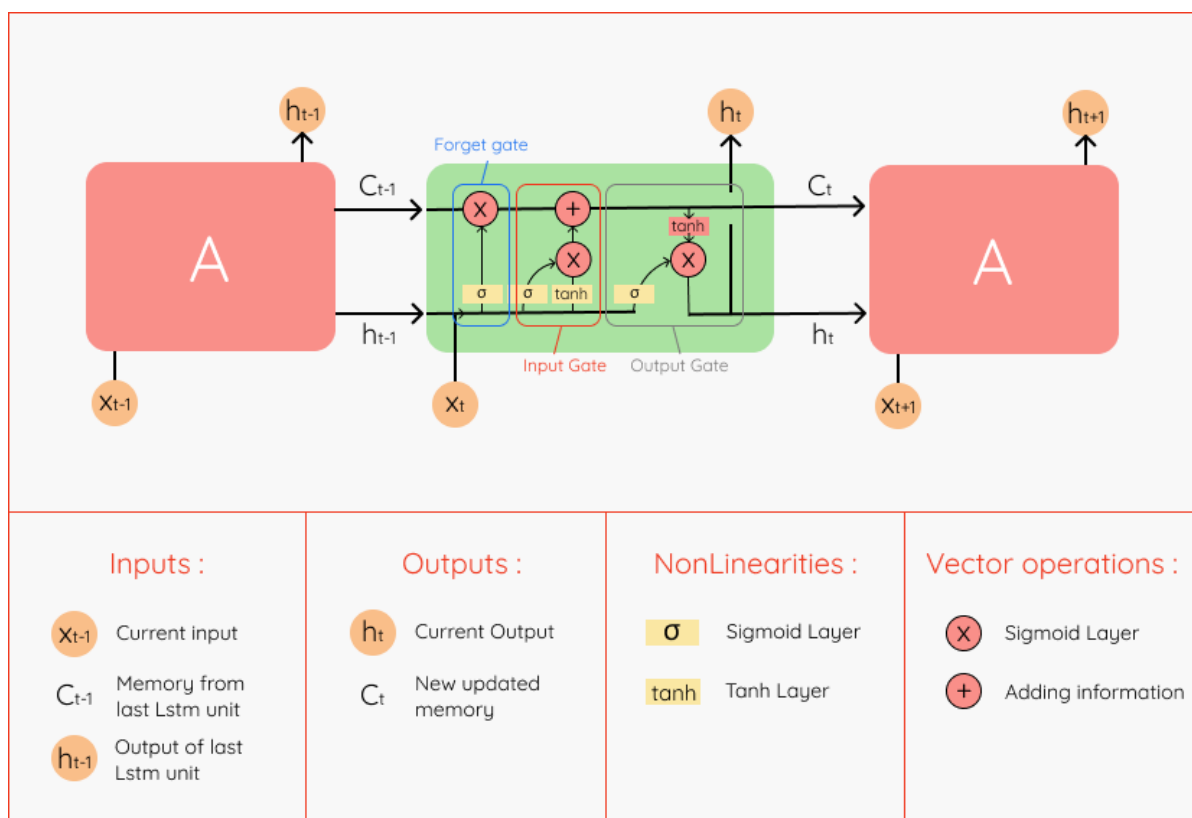


FIGURE 1.4 – Architecture de réseaux neuronaux récurrents avec mémoire à long terme (LSTM) [27].

Les unités de mémoire appelées cellules constituent un réseau LSTM ordinaire. L'état de la cellule et l'état caché sont tous deux transmis à la cellule suivante. La principale chaîne de flux de données - l'état de la cellule - permet aux données de se dérouler pratiquement sans modification. Il peut cependant y avoir quelques transformations linéaires. Grâce à des portes sigmoïdes, les données peuvent être ajoutées ou retirées de l'état de la cellule. Une porte est comparable à une couche ou à un ensemble de matrices qui contiennent chacune un ensemble différent de poids individuels. Comme les LSTM utilisent des portes pour réguler le processus de mémorisation, le problème de la dépendance à long terme est évité.

- La première étape de la construction d'un réseau LSTM consiste à identifier les informations qui ne sont pas requises et qui seront manquées de la cellule à cette étape. Ce processus d'identification et d'exclusion de données est décidé

par la fonction sigmoïde, qui prend la sortie de la dernière unité LSTM (h_{t-1}) au temps $t - 1$ et l'entrée courante X_t au temps t . De plus, la fonction sigmoïde détermine quelle partie de l'ancienne sortie doit être éliminée. Cette porte est appelée la porte de l'oubli (ou f_t); où f_t est un vecteur avec des valeurs allant de 0 à 1, correspondant à chaque nombre dans l'état de cellule, C_{t-1} .

$$f_t = \sigma(W_f[h_{t-1}, X_t] + b_f)$$

Ici, f_t est la fonction sigmoïde, et W_f et b_f sont les matrices de poids et le biais, respectivement, la porte du oubliez.

- L'étape suivante consiste à décider et à stocker les informations de la nouvelle entrée X_t dans l'état de cellule ainsi qu'à mettre à jour l'état de cellule. Cette étape contient deux parties, la couche sigmoïde et la seconde la couche \tanh . Tout d'abord, la couche sigmoïde décide si les nouvelles informations doivent être mises à jour ou ignorées (0 ou 1), et deuxièmement, la fonction \tanh donne du poids aux valeurs qui sont passées, en décidant de leur niveau d'importance (-1 à 1). Les deux valeurs sont multipliées pour mettre à jour le nouvel état de cellule. Cette nouvelle mémoire est ensuite ajoutée à l'ancienne mémoire C_{t-1} résultant en C_t .

$$i_t = \sigma(W_i[h_{t-1}, X_t] + b_i)$$

$$N_t = \tanh(W_n[h_{t-1}, X_t] + b_n)$$

$$C_t = C_{t-1} \cdot f_t + N_t \cdot i_t$$

Ici, C_{t-1} et C_t sont les états de cellule au temps $t - 1$ et t , tandis que W et b sont les matrices de poids et le biais, respectivement, de l'état de la cellule.

- Dans la dernière étape, les valeurs de sortie h_t sont basées sur l'état de la cellule de sortie O_t mais il s'agit d'une version filtrée. Tout d'abord, une couche sigmoïde décide quelles parties de l'état de la cellule parviennent à la sortie. Ensuite, la sortie de la porte sigmoïde O_t est multipliée par les nouvelles valeurs créées par la couche \tanh à partir de l'état de la cellule C_t , avec une valeur

comprise entre -1 et 1.

$$O_t = \sigma(W_o[h_{t-1}, X_t] + b_o)$$

$$h_t = O_t \cdot \tanh(C_t)$$

Ici, W_o et b_o sont les matrices de poids et le biais, respectivement, de la porte de sortie [27].

Réseau récurrent à portes (GRU)

Réseau récurrent à portes (Gated Recurrent Unit GRU) sont un type de réseau neuronal récurrent (RNN) qui a été introduit pour améliorer les réseaux LSTM traditionnels. Les GRU utilisent une seule porte de mise à jour et une porte de réinitialisation pour conserver ou oublier sélectivement les informations des entrées passées, ce qui leur permet de traiter des séquences d'entrée de longueur arbitraire et de maintenir un état qui encode des informations sur le passé. Les GRU sont plus simples et plus faciles à entraîner que les LSTM, tout en offrant des performances similaires dans de nombreuses tâches. Les GRU sont particulièrement bien adaptés aux tâches qui requièrent la capacité de se souvenir et d'utiliser des informations provenant de longues séquences, grâce à leur capacité à capturer des dépendances à longue portée dans des données séquentielles [28].

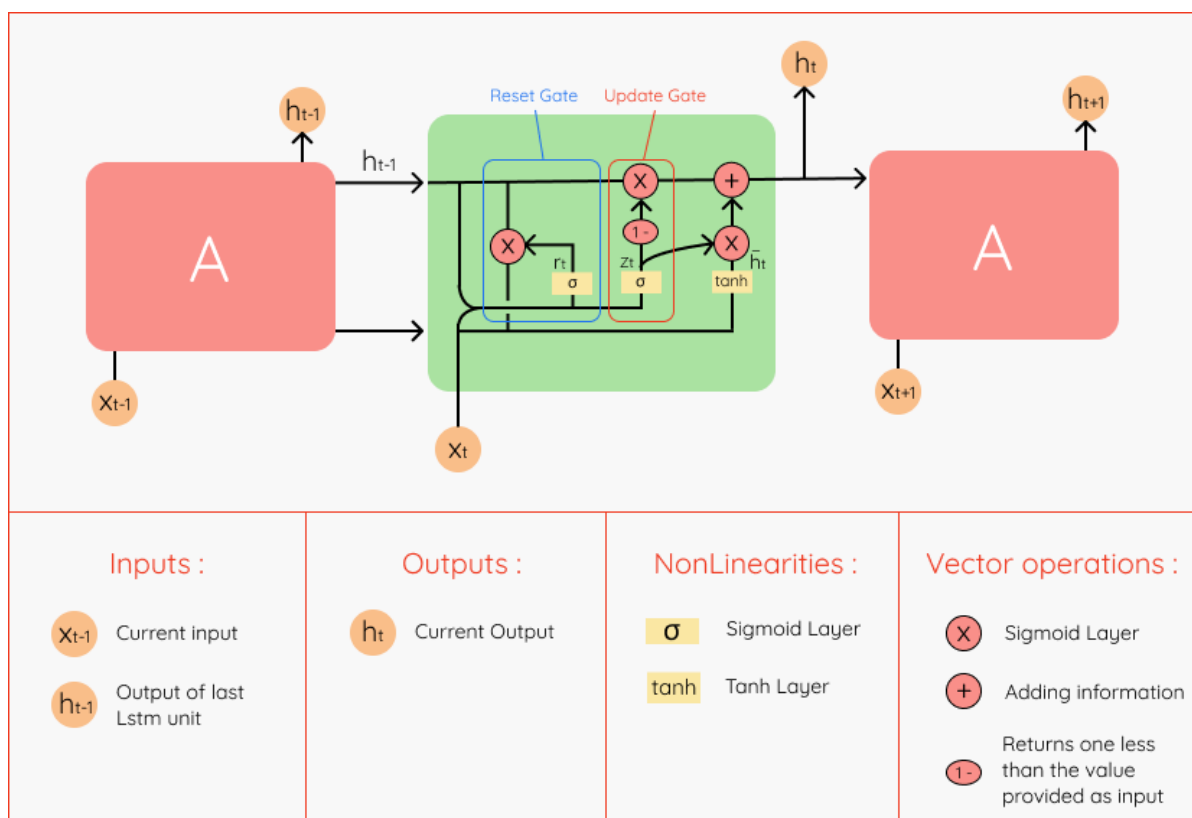


FIGURE 1.5 – Architecture de réseau récurrent à portes (GRU) [29].

La porte de réinitialisation (reset gate) r est utilisée pour sélectionner quelles informations doivent être ignorées de l'état caché précédent et des valeurs d'entrée ; Le rôle de la porte de mise à jour (update gate) z est de sélectionner quelles informations de l'état caché précédent doivent être conservées et transmises aux prochaines étapes ; La porte d'état candidate h_{t-1} calcule la valeur candidate pour l'état caché futur. Cela est réalisé en multipliant d'abord l'état précédent par la sortie de la porte de réinitialisation (reset gate). Cette étape peut être interprétée comme l'oubli des informations non pertinentes du passé. Ensuite, de nouvelles données provenant de l'entrée sont ajoutées aux informations restantes. Enfin, la fonction tangente hyperbolique (\tanh) est appliquée aux données pour réguler le flux d'informations. L'état caché actuel h_t est calculé comme suit. Tout d'abord, la sortie de la porte de mise à jour (update gate) z est soustraite de un, puis multipliée par l'état précédent h_{t-1} . Ensuite, la porte d'état

candidate g est multipliée par la sortie inchangée de la porte de mise à jour z . Les résultats de ces deux opérations sont finalement additionnés. Cela signifie que si les valeurs de sortie de la porte de mise à jour z sont proches de zéro, plus d'informations nouvelles sont ajoutées à l'état actuel h . En revanche, si les valeurs de sortie de la porte de mise à jour z sont proches de un, l'état actuel est principalement conservé tel qu'il était lors :

$$\begin{aligned} r(t) &= \sigma(W_r x(t) + R_r h(t-1) + b_r) \\ z(t) &= \sigma(W_z x(t) + R_z h(t-1) + b_z) \\ \tilde{h}(t) &= \tanh(W_{\tilde{h}(t)} x(t) + r(t) \odot (R_{\tilde{h}(t)} h(t-1)) + b_{\tilde{h}(t)}) \\ h(t) &= (1 - z(t)) \odot \tilde{h}(t) + z(t) \odot h(t-1) \end{aligned}$$

Lorsqu'on considère l'ensemble de la couche GRU avec n_N cellules, les matrices de poids $W_r, W_z, W_{\tilde{h}(t)}$ ont des dimensions $n_N \times n_f$, les matrices $R_r, R_z, R_{\tilde{h}(t)}$ ont des dimensions $n_N \times n_N$, et les vecteurs $b_r, b_z, b_{\tilde{h}(t)}$ ont des dimensions $n_N \times 1$ [30].

1.9 Conclusion

En conclusion, ce chapitre nous a permis de comprendre que l'apprentissage automatique est un domaine en constante évolution qui offre des opportunités pour automatiser des tâches, pour améliorer les performances et pour découvrir de nouvelles connaissances à partir de données existantes. Comme nous l'avons vu qu'il existe différents types d'apprentissage, y compris supervisé et non supervisé, ainsi que des techniques avancées telles que l'apprentissage en profondeur, qui peuvent aider à traiter des données volumineuses et complexes. Il est également important de noter que la qualité des données utilisées pour l'entraînement des modèles joue un rôle crucial dans les performances et les résultats obtenus.

CHAPITRE 2

CRYPTO-MONNAIES ET PRÉDICTION DE MARCHÉ

2.1 Introduction

Les crypto-monnaies sont des actifs numériques qui utilisent la cryptographie pour sécuriser les transactions et contrôler la création de nouvelles unités. Contrairement aux monnaies traditionnelles émises par les banques centrales et soutenues par les gouvernements, les crypto-monnaies sont décentralisées et fonctionnent sur un réseau peer-to-peer. Le bitcoin est la crypto-monnaie la plus connue, créée en 2009 par un individu ou un groupe inconnu sous le pseudonyme de Satoshi Nakamoto. Depuis, des milliers d'autres crypto-monnaies ont été créées, telles que l'Ethereum, le Litecoin et le Ripple, pour n'en citer que quelques-unes.

Les crypto-monnaies sont stockées dans des portefeuilles numériques, accessibles à l'aide de clés publiques et privées. Les transactions sont vérifiées et enregistrées sur un grand livre public appelé blockchain, maintenu par un réseau d'ordinateurs dans le monde entier. Cette décentralisation est l'une des caractéristiques clés des crypto-monnaies, ce qui signifie qu'elles ne sont pas contrôlées par une seule entité, telle qu'un gouvernement ou une institution financière. Certains partisans des crypto-monnaies soutiennent que cela est en fait une alternative plus démocratique

et transparente aux systèmes financiers traditionnels.

2.2 Définition

Un système de crypto-monnaie peut être considéré comme un système conçu pour émettre des jetons à des fins générales ou spécifiques en tant que moyen d'échange, et qui sont enregistrés à l'aide d'un grand livre commun fréquemment tenu à jour. L'utilisation du cryptage dans un grand livre numérique, à des degrés divers, remplace la confiance dans les institutions [31].

2.3 L'évolution des crypto-monnaies

L'histoire des crypto-monnaies est courte et il n'existe pas d'archives très complètes. Les monnaies numériques existaient avant les crypto-monnaies, mais elles ne sont pas identiques aux crypto-monnaies, qui sont décentralisées. Le bitcoin a été inventé par Satoshi Nakamoto en 2008. Il s'agit de la première forme décentralisée de monnaie numérique, sans organe central de gouvernance ou de contrôle. Depuis, de nombreuses autres crypto-monnaies décentralisées ont vu le jour. Le bitcoin est la crypto-monnaie la plus répandue et sa popularité a conduit à l'émergence de plus de 1000 autres crypto-monnaies, de nouvelles apparaissant fréquemment. Les crypto-monnaies ont prouvé leur valeur et leur capacité à fonctionner dans le monde réel, ce qui leur a valu d'être acceptées comme formes légitimes de monnaie et de paiement par les banques, les sociétés d'investissement, les organisations commerciales et les détaillants [32].

2.4 Trading de crypto-monnaie

Le trading de crypto-monnaies fait référence à l'achat et à la vente de crypto-monnaies sur des bourses ou par le biais de plateformes de trading dans le but de

réaliser un profit [33]. Les traders peuvent utiliser une variété de stratégies et de techniques pour prédire les mouvements du marché et prendre des décisions éclairées sur le moment d'acheter ou de vendre des crypto-monnaies spécifiques. Les traders de crypto-monnaies qui réussissent ont souvent une compréhension approfondie des tendances du marché et de l'analyse technique, ainsi qu'une volonté de prendre des risques calculés.

2.5 Avantages et inconvénients du trading des crypto-monnaies

Les avantages

- Les cryptomonnaies ont une volatilité élevée, ce qui peut offrir des opportunités aux investisseurs spéculateurs, mais implique également plus de risques que les investissements traditionnels.
- Le marché des crypto-monnaies est disponible 24 heures sur 24, 7 jours sur 7, et il est décentralisé, ce qui permet d'effectuer des transactions entre particuliers dans le monde entier.
- Les transactions en crypto-monnaies offrent un quasi-anonymat, ce qui peut permettre aux utilisateurs de bénéficier d'une plus grande confidentialité et d'une meilleure sécurité.
- Les bourses de crypto-monnaies disposent de mesures KYC spécifiques pour identifier les utilisateurs et réduire le risque financier tout en maximisant l'anonymat.
- Les transactions de pair à pair dans les crypto-monnaies peuvent réduire les coûts de transaction et attirer les utilisateurs qui se méfient des systèmes traditionnels.
- Certaines crypto-monnaies ont des capacités "intelligentes" programmables, y

compris des droits de propriété et de vote limités, et des participations partielles dans des actifs physiques tels que des œuvres d'art ou des biens immobiliers [34].

Les inconvénients

- Forte volatilité et risques importants : Investir dans les crypto-monnaies est très risqué en raison de leur forte volatilité. La valeur des crypto-monnaies peut fluctuer considérablement au quotidien, et tout événement ou déclaration lié à leur utilisation peut avoir un impact majeur sur leur demande et leur valeur sur le marché. C'est pourquoi les investisseurs doivent faire preuve de prudence et évaluer soigneusement les risques avant d'investir dans les crypto-monnaies. Ce point peut être considéré comme un avantage ou un inconvénient à la fois selon le degré de risque que l'on est prêt à prendre.
- La possibilité de perdre des crypto-monnaies : La responsabilité de la protection des portefeuilles de crypto-monnaies incombe uniquement à l'individu en raison de l'absence d'autorité centrale. La perte de la clé privée d'un portefeuille entraîne la perte définitive des crypto-monnaies qu'il contient, sans possibilité de les récupérer. Il est donc essentiel que les utilisateurs de crypto-monnaies prennent des mesures adéquates pour protéger leurs portefeuilles et leurs clés.
- Activités criminelles et cybersécurité : L'anonymat des crypto-monnaies permet aux criminels de les utiliser pour des activités illégales telles que le blanchiment d'argent, l'évasion fiscale et la vente non autorisée de drogues. La légalité des transactions effectuées avec des crypto-monnaies ne peut être vérifiée et n'importe qui peut posséder un portefeuille, y compris des criminels et des organisations terroristes. Des études montrent qu'un nombre important de transactions en bitcoins sont liées à des activités illégales et que le volume des transactions sur le dark-net utilisant des crypto-monnaies est en augmentation [35].

2.6 Prédiction du marché

La prédiction de marché est le processus d'estimation ou de projection de l'évolution future des marchés financiers, dans notre cas la valeur du Bitcoin. Les prédictions de marché sont basées sur l'analyse de différentes variables, telles que les données économiques, les tendances historiques, les modèles de prix, les facteurs fondamentaux et techniques, ainsi que les événements géopolitiques et économiques mondiaux.

Il est important de noter que la prédiction de marché comporte une part d'incertitude et qu'il est difficile de prévoir avec précision les mouvements futurs des marchés financiers. Les conditions économiques et les facteurs externes peuvent influencer les marchés de manière imprévisible.

2.6.1 Indicateurs

Les indicateurs présentés dans cette section sont basés sur l'étude des modèles et des tendances des prix passés d'un actif financier pour prévoir ses mouvements futurs. Ils se base sur l'hypothèse que les schémas de comportement des prix passés peuvent se répéter à l'avenir. Ces indicateurs sont aussi appelés "des facteurs techniques" qui peuvent inclure :

- Volume de transactions : Le volume des échanges de crypto-monnaies est une mesure de la quantité totale d'argent entrant et sortant d'un marché de crypto-monnaies au cours d'une période donnée. Tous les volumes provenant d'échanges contrôlés et décentralisés sont inclus dans le volume des transactions. Il s'agit d'une statistique cruciale utilisée par les traders et les investisseurs pour évaluer le sentiment du marché et estimer la valeur d'un certain actif cryptographique [36].
- Capitalisation : La capitalisation des cryptomonnaies, également connue sous le nom de "capitalisation boursière", représente la valeur actuelle du marché d'un réseau de cryptomonnaies. Cette valeur est obtenue en multipliant le prix

- d'une unité de compte par la quantité d'actifs crypto en circulation sur le marché [37].
- Moyenne mobile : Une moyenne mobile (MM) est un outil d'analyse technique largement utilisé en finance pour lisser les données de prix en calculant un prix moyen actualisé sur une période de temps donnée. Le calcul de la moyenne mobile d'une action a pour but de filtrer les fluctuations de prix à court terme et de mettre en évidence la tendance sous-jacente de l'action. Les moyennes mobiles simples (SMA) calculent la moyenne arithmétique des prix sur une période donnée, tandis que les moyennes mobiles exponentielles (EMA) accordent plus d'importance aux prix récents qu'aux prix plus anciens sur la même période [38].
 - Support et résistance : Le support et la résistance sont des concepts clés de l'analyse technique utilisés pour identifier les niveaux de prix potentiels où des pressions d'achat ou de vente peuvent s'exercer sur le marché. Le support est un niveau ou une zone sur un graphique de prix où les acheteurs sont disposés à acheter et où la demande est susceptible de dépasser l'offre, ce qui fait que les prix cessent de baisser et s'inversent. La résistance, quant à elle, est un niveau ou une zone où les vendeurs sont disposés à vendre et où l'offre est susceptible de dépasser la demande, ce qui fait que les prix cessent d'augmenter et s'inversent. Ces niveaux peuvent être trouvés dans toutes les périodes graphiques et peuvent servir de points d'entrée ou de sortie potentiels pour les traders. L'analyse technique n'est pas une science exacte et il faut faire preuve de souplesse dans l'interprétation des graphiques [39].
 - RSI (Relative Strength Index) : en français, l'indice de force relative, est l'un des indicateurs les plus utilisés sur les marchés financiers pour mesurer la force et la dynamique des mouvements de prix dans l'analyse technique. Il est un outil qui aide les utilisateurs de l'analyse technique à comprendre et interpréter les graphiques techniques, leur permettant de prédire les tendances futures des prix sur les marchés financiers et de mesurer l'amplitude des mouvements de

prix. Le RSI mesure la comparaison entre les tendances à la hausse et à la baisse sur une période spécifique, attribuant des valeurs allant de 0 à 100. Il est calculé en divisant les prix de clôture moyens des périodes de hausse par ceux des périodes de baisse, en utilisant généralement une période de 14 jours. Le RSI est également utile pour identifier les conditions de surachat et de survente, avec des seuils fixés à 30 et 70. Lorsque le RSI passe en dessous de 30 puis remonte, cela indique un marché suracheté avec une poursuite de la tendance à la hausse. En revanche, lorsque le RSI dépasse 70 puis redescend, cela suggère un marché survendu avec une poursuite de la tendance à la baisse [40].

2.6.2 Difficultés

Les indicateurs vus dans la section précédente peuvent donner une idée sur la tendance du marché pour une cryptomonnaie (ou autre actif), néanmoins, la prédiction exacte de sa valeur reste impossible pour plusieurs raisons, nous citons :

- Volatilité : La volatilité est une mesure qui quantifie le degré de fluctuation du prix d'un actif sur une période donnée. En règle générale, les actifs à forte volatilité sont perçus comme des investissements plus risqués, mais ils sont également susceptibles de générer des profits ou des pertes plus importants sur des périodes plus courtes que les actifs relativement moins volatils [41].
- Manque de réglementation : L'absence de réglementation claire pour les cryptomonnaies et les tentatives des gouvernements de les classer comme des titres, des monnaies ou les deux créent un environnement d'incertitude pour les investisseurs. Il peut en résulter des difficultés pour vendre les crypto-monnaies et une baisse significative de leur valeur en cas de répression réglementaire soudaine. En outre, ce manque de surveillance réglementaire peut également rendre les crypto-monnaies vulnérables à la fraude, au piratage et à d'autres activités malveillantes qui peuvent nuire davantage aux investisseurs [42].
- Spéculation et comportement des investisseurs : La spéculation implique des transactions financières à haut risque visant des gains significatifs, malgré des

pertes potentielles. Les spéculateurs se concentrent sur les fluctuations de prix à court terme, motivés par le désir de réaliser des bénéfices importants. Cette nature spéculative rend la prédiction des cryptomonnaies difficile en raison de l'accent mis sur la volatilité des prix à court terme plutôt que sur la valeur à long terme, ce qui rend difficile de prévoir avec précision leur comportement en se basant sur les principes d'investissement traditionnels [43].

- Manipulation du marché : La manipulation du marché consiste à influencer le prix d'un actif ou le comportement du marché à des fins de gain personnel, souvent par des moyens contraires à l'éthique tels que la diffusion de fausses informations ou la création d'une demande ou d'une offre fictive. Un exemple de manipulation du marché des crypto-monnaies est le "wash trading". Cette stratégie consiste pour une personne ou un groupe à acheter et à vendre rapidement la même crypto-monnaie pour en gonfler artificiellement le volume afin de gagner l'intérêt des investisseurs [44].

2.7 Travaux connexes

L'analyse et la prévision des prix des crypto-monnaies est un sujet d'étude très difficile et un défi très exigeant dans l'analyse des séries temporelles. Les énormes changements et la volatilité des séries temporelles des crypto-monnaies, qui sont fortement influencées par un immense nombre d'événements, sont la source de sa complexité et de sa difficulté. Afin d'augmenter la précision des prévisions, des initiatives de recherche récentes ont intégré et utilisé des approches d'apprentissage profond pour prévoir le prix du bitcoin et le mouvement directionnel. Nous présentons brièvement quelques résultats intrigants et des conclusions perspicaces.

- L'étude de Hamayel et Owda [45] ont examiné l'utilisation de trois algorithmes récurrents de réseaux neuronaux, à savoir le GRU (Gated Recurrent Unit), le LSTM (Long Short-Term Memory) et le bi-LSTM (LSTM bidirectionnel), pour prédire les prix du Bitcoin (BTC), du Litecoin (LTC) et de l'Ethereum (ETH).

Les données utilisées dans l'étude comprenaient les variables : open , high, low et close.

Les résultats obtenus ont montré que le modèle LSTM avait un RMSE de 410.399 et un MAPE de 1.1234% pour la prédiction des prix du Bitcoin. Le modèle bi-LSTM a obtenu des résultats légèrement moins précis, avec un RMSE de 2927.006 et un MAPE de 5.990%. En revanche, le modèle GRU a présenté les meilleures performances, affichant un RMSE de 174.129 et un MAPE de 0.2454%.

- Dans l'étude de Rizwan et al [46], les auteurs ont comparé les performances de trois modèles proposés : RNN, LSTM et GRU. Ils ont trouvé que le nombre total de paramètres et la taille de l'ensemble de données peuvent significativement impacter la précision des modèles. Les résultats ont montré que le modèle GRU a surpassé le modèle LSTM, atteignant une précision optimale de 94,70% et une amélioration de 42,3% par rapport aux modèles RNN et LSTM précédents. Cependant, les auteurs ont noté que les caractéristiques sélectionnées telles que Open , High, Low et Close n'étaient pas suffisantes pour prédire avec précision la valeur du Bitcoin. Ils ont donc recommandé de recueillir et d'intégrer des informations supplémentaires pour améliorer la précision des modèles.
- Alvin HO et al proposent dans [47] une étude qui révèle que le meilleur taux de précision est atteint par le modèle de mémoire à long terme (LSTM) par rapport à la régression linéaire. Alors que la régression linéaire s'appuie sur l'équation quadratique moyenne pour mesurer la précision sur la base du graphique linéaire de l'ensemble de données en temps continu, elle a atteint un taux de précision impressionnant d'environ 99,97% pour les données d'apprentissage et de test. D'autre part, le modèle LSTM utilise l'erreur absolue moyenne comme mesure pour évaluer la précision, et a obtenu un taux d'erreur d'environ 0,08%. Il est important de noter que cette étude s'est concentrée exclusivement sur les caractéristiques des prix Open , High , low et cClose. Toutefois, compte tenu de la nature volatile du marché des crypto-monnaies, qui est influencé par des facteurs externes tels que les médias sociaux et les actualités, il est essentiel de

reconnaître que les prévisions ne peuvent pas s'appuyer uniquement sur des ensembles de données.

- L'étude de Abdullah H. AL-NEFAIE et Theyazn H. H. ALDHYANI [48] a exploré l'utilisation de modèles d'intelligence artificielle pour prédire les prix du bitcoin, en se concentrant sur les modèles GRU et MLP. Les résultats ont montré que le modèle MLP a surpassé le modèle GRU en termes de performances, avec des résultats de MSE (erreur quadratique moyenne) de 0,03354 et de RMSE (racine carrée de l'erreur quadratique moyenne) de 0,18314 pour le modèle GRU, et de MSE : 0,000109 et RMSE : 0,01044 pour le modèle MLP. Ces résultats démontrent la capacité des modèles d'apprentissage profond à fournir des estimations précises de la valeur réelle du marché du Bitcoin. De plus, ces modèles permettent une évaluation plus précise du timing des transactions, ce qui peut avoir un impact significatif sur l'économie en influençant les décisions des investisseurs et des commerçants. Cette étude met en évidence l'importance croissante de l'utilisation de l'intelligence artificielle pour prédire les prix des cryptomonnaies et souligne la nécessité de développer des modèles plus précis pour faire face aux défis posés par la volatilité des devises numériques.
- Dans l'étude menée par Seabe PL et al [49] sur la prédiction des prix des cryptomonnaies, trois modèles ont été testés : GRU, LSTM et Bi-LSTM, pour prédire les prix de trois cryptomonnaies, à savoir Bitcoin (BTC), Litecoin (LTC) et Ethereum (ETH). Les données utilisées comprenaient : Date , Open, High, Low, Close et adjusted close.

En ce qui concerne les résultats spécifiques fournis pour le Bitcoin, le modèle LSTM a obtenu une erreur de pourcentage absolue moyenne (MAPE) de 0.039 et une erreur quadratique moyenne (RMSE) de 1031.340. Le modèle Bi-LSTM a légèrement surpassé le modèle LSTM, avec une MAPE de 0.036 et un RMSE de 1029.362. En revanche, le modèle GRU a présenté des valeurs d'erreur plus élevées, avec une MAPE de 0.057 et un RMSE de 1274.171.

Ces résultats suggèrent que les modèles LSTM et Bi-LSTM ont obtenu de meilleures performances en termes de précision par rapport au modèle GRU.

- l'étude réalisée par Pasak S et Jayadi R [50] s'est concentrée sur la prédiction des prix des cryptomonnaies en utilisant les modèles ARIMA et LSTM. Plusieurs cryptomonnaies, dont Bitcoin, Ethereum, Binance Coin, Tether et Cardano. Les caractéristiques des données comprenaient Date, Open, High, Low, Close, Adjuste Close et Volume. Les résultats de l'étude ont montré que le modèle LSTM surpassait le modèle ARIMA (AutoRegressive Integrated Moving Average) en termes de précision et de capacité de prévision. Pour Bitcoin, le modèle LSTM a obtenu un MAPE de 5,202 et un RMSE de 2879,051, tandis que le modèle ARIMA a obtenu un MAPE de 6,952 et un RMSE de 0,797. Alors que la méthode ARIMA était capable de capturer les variations et de mettre en évidence les tendances et les saisons, la méthode LSTM était plus efficace pour interpréter les mouvements des données et suivre plus précisément les fluctuations des prix des cryptomonnaies.
- Akila V et al. [51] ont concentré sur la prédiction des prix des cryptomonnaies en utilisant les algorithmes LSTM et PELT avec des données de séries chronologiques. Leur méthodologie proposée visait à exploiter les algorithmes d'apprentissage profond pour capturer les modèles complexes dans les données de séries chronologiques, tout en incorporant des techniques de détection des points de changement pour s'adapter aux changements soudains dans les données. Les résultats expérimentaux ont montré que le réseau LSTM avec l'algorithme PELT a surpassé le modèle LSTM de base en termes de toutes les mesures d'évaluation. Surtout là est une amélioration significative de 30% de la valeur de RMSE lorsque le modèle LSTM+PELT
- Dans l'étude de Panggabean R et Widayarsi Y. D. L [52] plusieurs expériences ont été effectuées en utilisant différents modèles tels que SVR, Random Forest, LSTM et GRU pour prédire le prix du Bitcoin. Ils ont utilisé un ensemble de

données comprenant les caractéristiques suivantes : Open, High, Low, Close , Adjuste Close et Volume. Le modèle GRU a donné les meilleurs résultats avec les métriques suivantes : RMSE : 4159.41, MSE : 191.41, MAE : 2673.46.

L'élément commun entre toutes ces études et d'autres, c'est qu'ils suggèrent que l'analyse technique de l'historique des valeurs des cryptomonnaies ne serait en aucun cas suffisant pour prédire leurs valeurs à moyen et/ou long terme. Plusieurs autres facteurs sont susceptibles d'affecter les prix du marché des cryptomonnaies, tels que les médias sociaux, le sentiment du public, la régulation des cryptomonnaies ainsi que les monnaies traditionnelles, etc. Malheureusement ces facteurs sont très compliqués voire impossibles à modéliser avec précision pour qu'un modèle mathématique puisse les analyser.

2.8 Conclusion

Les crypto-monnaies sont apparues comme une nouvelle forme de monnaie numérique qui a bouleversé le système financier traditionnel. L'évolution des crypto-monnaies a été rapide et leurs avantages, tels que la décentralisation, la rapidité des transactions et la faiblesse des frais, ont attiré de nombreux investisseurs et négociants. Toutefois, leurs inconvénients, tels que la volatilité, les problèmes de sécurité et la possibilité de les utiliser dans le cadre d'activités criminelles, ne peuvent être ignorés. Le trading de crypto-monnaies est une activité très complexe et difficile, avec des risques et des incertitudes importants. L'apprentissage automatique a été utilisé pour développer des algorithmes de négociation afin d'atténuer certains de ces risques, mais cela reste un domaine difficile et dynamique qui nécessite une recherche et un développement continus. Dans l'ensemble, les crypto-monnaies sont un domaine complexe et évolutif qui nécessite un examen attentif et une attention particulière de la part des investisseurs et des régulateurs afin de garantir leur utilisation sûre et responsable.

CHAPITRE 3

CONCEPTION

3.1 Introduction

Dans ce chapitre, nous présentons un modèle MLP (Multi-Layer Perceptron) pour la prévision des cryptomonnaies. Le modèle précédent ne prend en compte qu'un seul jour pour prédire la valeur du jour suivant, le modèle que nous proposons vise à prédire les prix des cryptomonnaies sur des périodes pouvant s'étendre sur plusieurs jours ou mois.

3.2 Architecture générale du système

L'objectif de notre travail est de concevoir un système intelligent capable de prédire le prix des cryptomonnaies. Pour commencer, nous avons collecté un ensemble de données sur le Bitcoin sur lequel nous souhaitons travailler. Nous avons ensuite effectué un prétraitement sur ces données, comprenant la vérification des données manquantes et la normalisation des données ··· etc.

Une fois le prétraitement terminé, les données ont été fournies à notre modèle pour l'apprentissage. Après l'apprentissage, le modèle est utilisé pour effectuer des prédictions.

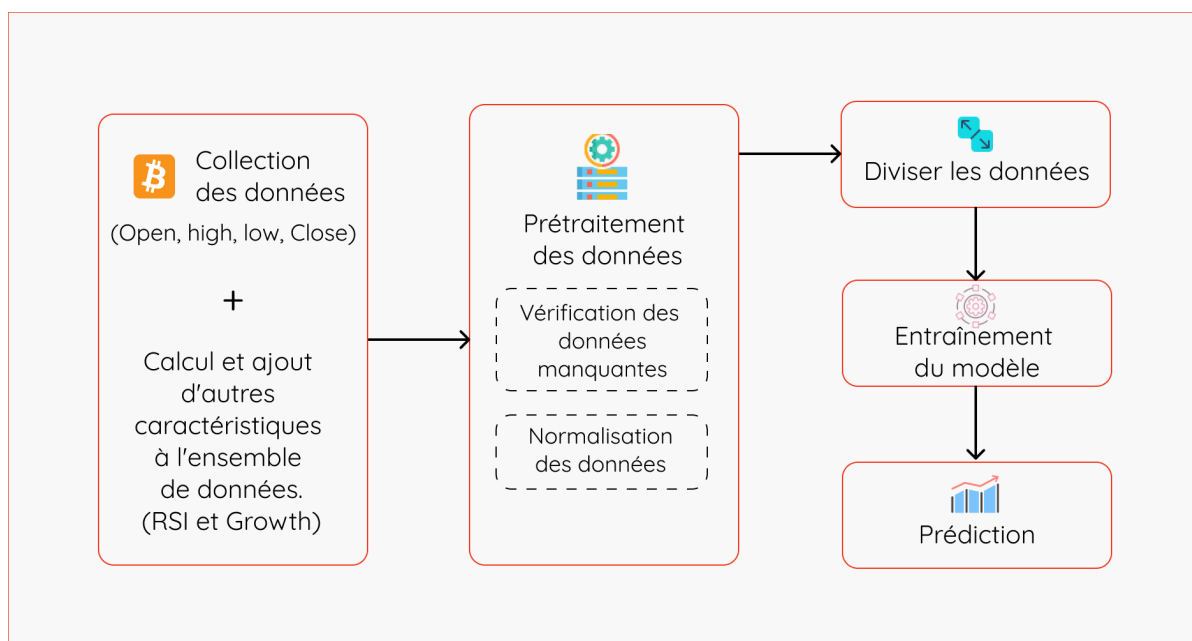


FIGURE 3.1 – Architecture générale du système

3.2.1 La collection de l'ensemble de données

Dans cette étude, nous avons utilisé un ensemble de données provenant de sources fiables, comprenant les prix Ouverture, Plus haut, Plus bas, Clôture OHLC (Open-High-Low-Close) quotidiens. Cependant, nous allons utiliser uniquement la valeur de clôture (Close) et l'ouverture (Open) pour notre étude. Nous avons également calculé l'indice de force relative RSI et la Croissance (Growth) de manière autonome, d'autres indices aussi ont été calculés, mais nous n'allons pas les aborder ici car ils n'ont pas été retenus sur la version finale de notre modèle.

OHLC (Open-High-Low-Close)

L'OHLC désigne les prix d'ouverture, de clôture, les prix hauts et bas pour chaque période. Les OHLC sont utiles car ils montrent les quatre principaux points de données sur une période, le prix de clôture étant considéré comme le plus important par de nombreux traders, il est considéré comme référence pour le prix de la cryptomonnaie dans une unité de temps.

- Open : Le prix de première transaction pour une crypto-monnaie donnée à le début d'un échange quotidien.
- Close : Le prix de la dernière transaction pour une crypto-monnaie donnée à la fin d'un échange quotidien.
- High : Le prix le plus élevé payé pour une crypto-monnaie au cours d'un échange quotidien.
- Low : Le prix le plus bas d'une crypto-monnaie atteint lors d'un échange quotidien [53].

Calcul du RSI

Pour calculer le RSI, nous commençons d'abord par le calcul du RS (Relative Strength - Force Relative), qui est donné par la moyenne des gains, par la moyenne des pertes. Pour un jour donné, si la différence entre le prix de fermeture actuel et celui de la journée précédente (ou la différence entre le prix d'ouverture et celui de fermeture). Si cette différence est positive, alors nous avons un gain à l'issue de cette période, sinon nous avons une perte.

Pour le calcul du RS, nous considérons séparément la somme des gains et des pertes, puis nous divisons les gains par les pertes (ou nous divisons leurs moyennes car le résultat et le même).

$$RS = \frac{\text{Moyenne des gains}}{\text{Moyenne des pertes}} \quad (3.1)$$

Par la suite, le calcul du RSI se fait à l'aide de la formule suivante :

$$RSI = 100 - \frac{100}{1 + RS} \quad (3.2)$$

Notez ici que le RSI tend vers 100 lorsque la valeur de RS tend vers l'infini (croissance continue sur 14 pas de temps dans notre cas), et tend vers zéro si la valeur de RS tend également vers zéro (décroissance sur 14 jours) [40]. Même si théoriquement, la

valeur du RSI peut varier entre zéro et 100, en pratique, nous observons très rarement des valeurs supérieures à 90 ou inférieures à 10.

L'indice de force relative peut être calculé pour n'importe quel pas de temps, nous avons expliqué comment le calculer pour un pas de temps en jours, mais la même logique est appliquée pour calculer le RSI des 14 dernières semaines, mois, heures ou minutes. Dans le cadre de notre travail, nous calculons le RSI pour un pas d'une journée, 3 jours, une semaine et un mois, nous appelons ces nouvelles caractéristiques : RSI1, RSI3, RSI7 et RSI30.

- RSI1 représente le RSI quotidien. Cela signifie que les prix de clôture utilisés dans le calcul représentent uniquement le prix de clôture de cette journée spécifique.
- RSI3 représente le RSI calculé sur une période de 3 jours. Dans ce cas, le prix de clôture du jour actuel représente les prix de clôture des trois derniers jours.
- RSI7 représente le RSI calculé sur une période de 7 jours. Le prix de clôture du jour actuel représente les prix de clôture des sept derniers jours.
- RSI30 représente le RSI calculé sur une période de 30 jours. Le prix de clôture du jour actuel représente les prix de clôture des trente derniers jours.

Chaque intervalle RSI offre une perspective différente sur la force relative ou la faiblesse de l'actif analysé, en fonction de la période choisie. Cela permet d'examiner les tendances à court terme et à long terme, et aide à identifier les éventuelles conditions de surachat ou de survente sur le marché.

Finalement, nous avons calculé les valeurs de growth en utilisant la formule suivante :

$$\text{Growth} = (\text{close} - \text{open}) \div \text{open} \times 100 \quad (3.3)$$

3.2.2 Prétraitement de données

Dans l'ensemble de données, nous avons vérifié les valeurs manquantes et n'en avons trouvé aucune. Nous n'avons pas non plus supprimé les valeurs aberrantes,

car les prix des bitcoins sont influencés par divers facteurs et les valeurs aberrantes peuvent représenter de véritables fluctuations du marché. Leur suppression peut entraîner la perte d'informations importantes. Par conséquent, nous avons utilisé la normalisation uniquement comme méthode de traitement des données.

3.2.3 Organiser et Diviser l'ensemble données

Après le prétraitement, nous allons diviser les données en un ensemble d'entraînement et un ensemble de test.

Avant de procéder à la division, nous avons commencé par utiliser le jour actuel comme entrée, associé à la valeur de "Growth" correspondante du jour suivant en tant que valeur cible, pour chaque instance de données. Par conséquent, nous perdrons le dernier jour des données cibles, que nous remplacerons uniquement par la valeur 0. Finalement, nous avons divisé les données en ensembles distincts (entraînement et test).

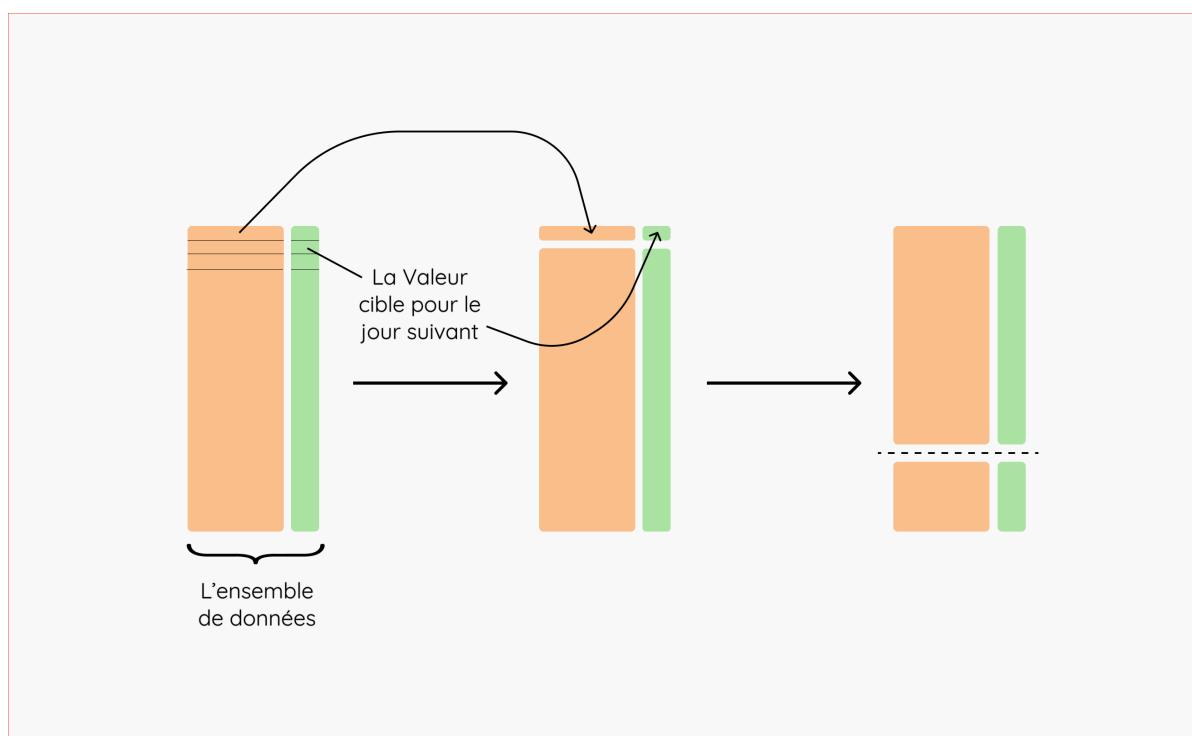


FIGURE 3.2 – Organiser et diviser l'ensemble de données

3.2.4 Phase d'apprentissage

Après le prétraitement et la division des données, les données sont passées au modèle proposé pour apprentissage. Les modèles proposés est MLP (Multi-Layer Perceptron). La taille des lots et le nombre d'époques peuvent également être utilisés. La taille du lot détermine le nombre d'échantillons traités en une seule fois, tandis que le nombre d'époques détermine le nombre de fois que le modèle passe sur l'ensemble des données.

Il est plus raisonnable d'utiliser d'utiliser un modèle de type LSTM ou même RNN ou GRU en raison de leur capacité à traiter des données temporelles, nous allons prendre un chemin différent en utilisant un réseau de neurones classique, et l'entraîner sur des données qui comporte une information temporelle. Notre modèle prend en entrée les valeurs RSI1, RSI3, RSI7 et RSI30, le modèle n'a aucune information sur la valeur actuelle de la cryptomonnaie, il est entraîné à calculer une prédiction de la croissance, la valeur de fermeture de la journée sera calculée à l'aide de la valeur d'hier et le taux de croissance (ou décroissance) prédit par le modèle.

3.2.5 Prédiction

Après l'entraînement et la sauvegarde du modèle, nous avons effectué une prédiction en utilisant les données de test. Étant donné que notre modèle prédit les valeurs de "croissance", nous utiliserons cette prédiction pour calculer la valeur de "fermeture". De plus, nous calculerons les valeurs RSI1, RSI3, RSI7 et RSI30 pour chaque prédiction (les RSI1, RSI3, RSI7 et RSI30 sont calculés de la même manière que pour la création de l'ensemble des données). Il est important de calculer ces caractéristiques car nous les utiliserons pour la prochaine prédiction.

Nous avons observé que dans certains cas, notre modèle génère des séquences croissantes (ou décroissantes) trop longues pour être réalistes, pour cela nous avons aussi introduit une deuxième vérification suite à la prédiction du modèle. Si nous obtenons un taux de croissance positif en présence de RSI supérieur à 70, ou un taux de

croissance négatif en présence de $RSI < 30$, une correction est introduite à la prédiction donnée par le modèle. Cela se fait en ajoutant une fonction $f(RSI)$ à la prédiction. La croissance corrigée est calculée à l'aide de la formule suivante :

$$corrig = prdiction \times (1 - f(RSI)) \quad (3.4)$$

avec la fonction $f(RSI)$ qui retourne une valeur entre 0 et 2. Si $f(RSI) > 1$, la prédiction est inversée (passe de positive à négative ou l'inverse), sinon la valeur de la prédiction est réduite sans changer sa direction. Une fois la prédiction terminée, nous enregistrons les résultats. Ensuite, nous pouvons utiliser des mesures de performance pour évaluer notre modèle.

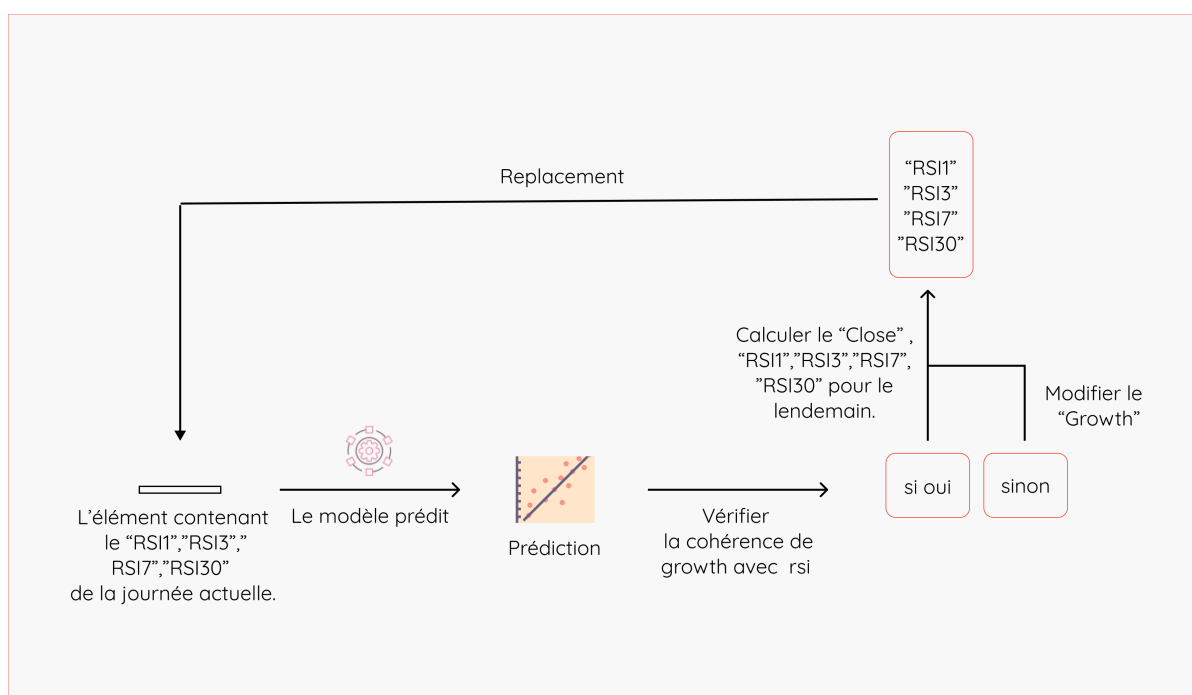


FIGURE 3.3 – Processus de prédiction

3.3 Conclusion

pour construire le système de prédiction de prix de Bitcoin, nous sommes passés par trois étapes : la phase de prétraitement des données initiales pour les préparer,

la phase d'apprentissage pour entraîner le modèle de prédiction afin d'améliorer les résultats, et enfin la phase de prédiction utilisant des données de test pour tester le modèle. Le modèle de prédiction proposé est le MLP(Multi-Layer Perceptron).

CHAPITRE 4

IMPLÉMENTATION ET RÉSULTATS

4.1 Introduction

Ce chapitre est structuré en deux sections principales. La première section se concentre sur l'implémentation du modèle proposé et discute des choix de l'implémentation effectués, y compris l'environnement de développement, les bibliothèques utilisées et les résultats de l'ajustement des paramètres. La deuxième section de ce chapitre explore les différents tests effectués et les résultats obtenus.

4.2 Implémentation

Dans cette section, nous fournissons des détails sur l'environnement de développement et le langage de programmation utilisés pour créer notre système. Nous présentons également les bases de données d'apprentissage et de test utilisées, ainsi que les détails de l'implémentation de l'architecture proposée et de l'interface graphique de l'application.

4.2.1 Environnement de développement

L'implémentation de notre système a été réalisée sur une machine possédant les caractéristiques suivantes avec les logiciels présentés ci-dessous :

- Processeur : Intel® Core™ i5-8350U CPU @1.7GHZ 1.90GHZ.
- Mémoire :8.00 Go DDR4.
- Disque dur : 256 GB.
- système d'exploitation : Windows 10 Professionnel.

4.2.2 Langage de programmation et bibliothèques

Python 3.9.12

Est un langage de programmation robuste et facile à apprendre. Il offre des structures de données de haut niveau qui sont efficaces, et son approche de la programmation orientée objet est simple mais efficace. La nature interprétée du langage, sa syntaxe élégante et son typage dynamique le rendent adapté à l'écriture de scripts et au développement rapide d'applications dans des domaines variés et sur des plateformes multiples [54].

Bibliothèques utilisées

- TensorFlow : est une plateforme open source de bout en bout pour l'apprentissage automatique ,qui permet aux débutants et aux experts de créer facilement des modèles d'apprentissage automatique [55]. nous avons utilisé cette bibliothèque pour définir les composants de base de l'architecture.
- Keras : permet les bibliothèques utilisées avec TensorFlow est Keras, nous avons utilisé cette bibliothèque pour de créer et d'entraîner des modèles de deep learning [56].

- Sklearn : Scikit-learn est une bibliothèque d'apprentissage automatique libre et gratuite conçue pour simplifier la mise en œuvre de divers algorithmes d'apprentissage automatique, y compris l'apprentissage supervisé et non supervisé. Elle fournit une large gamme d'outils pour des tâches telles que le prétraitement des données, la sélection des modèles et l'évaluation des modèles. Ces outils aident les développeurs à construire et à tester rapidement leurs modèles d'apprentissage automatique, ce qui permet une analyse efficace et précise de vastes ensembles de données [57].
- NumPy : une bibliothèque Python pour le calcul scientifique qui fournit des outils pour manipuler des tableaux multidimensionnels et effectuer diverses opérations sur ceux-ci, telles que des opérations mathématiques, des tris, des sélections et des analyses statistiques. Elle comprend également des outils pour l'algèbre linéaire de base, les transformées de Fourier discrètes et la simulation aléatoire [58].
- Pandas : est un outil open-source d'analyse et de manipulation de données qui est rapide, puissant, flexible et facile à utiliser. Il est basé sur le langage de programmation Python [59] et est largement utilisé pour son efficacité dans le traitement de grands ensembles de données.

4.2.3 Implémentation de l'architecture proposé

La collection de l'ensemble de données

Les données quotidiennes ont été collectées sur une période de 8 ans et 8 mois, allant du 17 septembre 2014 au 08 juin 2023, ce qui nous donne une vue d'ensemble du marché du Bitcoin et de ses tendances pendant cette période spécifique.

Les données OHLC ont été obtenues à partir de la source de données mentionnée dans la référence [60].

Prétraitement de données

Après s'être assuré qu'il n'y a pas de valeurs manquantes dans l'ensemble de données, nous avons appliqué une technique de normalisation appelée "MinMax-Scaler" pour modifier les valeurs des colonnes numériques de notre ensemble de données. L'objectif de la normalisation est de mettre toutes les valeurs à une échelle commune, sans altérer les différences dans les plages de valeurs. Cette opération accélère l'apprentissage et permet une convergence plus rapide du modèle.

Diviser les données

Dans notre travail, nous avons divisé les données en utilisant une répartition de 80% pour l'apprentissage et 20% pour les tests. Cette répartition équitable permet d'avoir suffisamment de données pour l'entraînement du modèle, tout en réservant une quantité raisonnable pour évaluer ses performances. En utilisant 80% des données pour l'entraînement, le modèle est en mesure d'apprendre de manière plus efficace les motifs et les relations présents dans l'ensemble de données. Les 20 % restants sont réservés pour tester les performances du modèle entraîné. Cet ensemble de tests séparé permet d'évaluer la capacité du modèle à généraliser aux données non vues et à corriger d'éventuelles erreurs.

Phase d'apprentissage

- L'importation des bibliothèques nécessaires Tensorflow, Sklearn, Keras, Numpy, plotly, matplotlib, etc.
- L'importation des bases des données dataset.csv avec la fonction `read_csv()`.
- Construire le modèle
- L'évaluation de modèle.
- Sauvegarder le modèle obtenu.

Pour avoir les meilleurs paramètres qui conduisent aux bons résultats nous avons relancé l'apprentissage plusieurs fois. Les paramètres principaux a fixé pour l'apprentissage sont :

- Batch-size : paramètre spécifie le nombre d'échantillons utilisés dans chaque mise à jour des paramètres du modèle. Cela peut affecter la vitesse d'entraînement et l'utilisation de la mémoire. Ici, il est défini sur 10, ce qui signifie que le modèle mettra à jour ses paramètres après avoir traité 10 échantillons.
- Epoch : est le nombre total d'itérations d'apprentissage, il est défini comme un critère d'arrêt que ce soit les résultats. Nous l'avons fixé à 10 epochs.
- fonction d'activation : En utilisant "ReLU" dans la première couche, votre modèle peut apprendre des représentations non linéaires complexes des caractéristiques d'entrée, tandis que l'activation "linear" dans la couche de sortie facilite la prédiction directe de la croissance continue des prix sans aucune limitation.
- Optimiseur : Les optimiseurs sont des algorithmes ou des méthodes utilisés pour modifier les attributs de votre réseau de neurones tels que les poids et le taux d'apprentissage afin de réduire les pertes.

4.2.4 Interface du système

Le modèle proposé a été enrichi d'une interface graphique conviviale pour faciliter son utilisation. Lorsque l'utilisateur lance l'application, une fenêtre principale s'affiche, comportant quatre boutons fonctionnels. Les deux premiers boutons permettent de sélectionner une date de début et une date de fin, tandis que le troisième bouton déclenche la prédiction. Le quatrième bouton ouvre une fenêtre dans le navigateur qui affiche un graphique avec des bougies (candles) et des lignes représentant les valeurs de clôture en fonction de leur date. De plus, cette fonctionnalité offre la possibilité de suivre la valeur et le jour correspondant en survolant les lignes et les bougies.

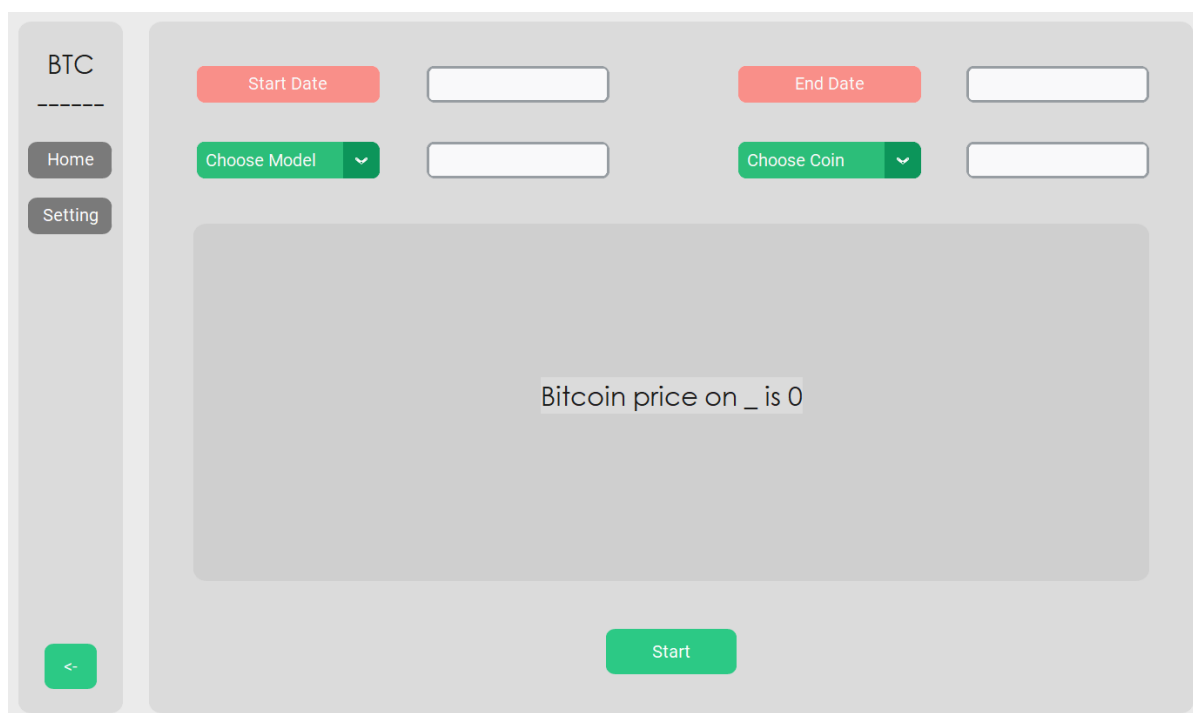


FIGURE 4.1 – La fenêtre principale de l'application

Les boutons "Start Date" et "End Date" ouvrent une petite fenêtre permettant de sélectionner la date.

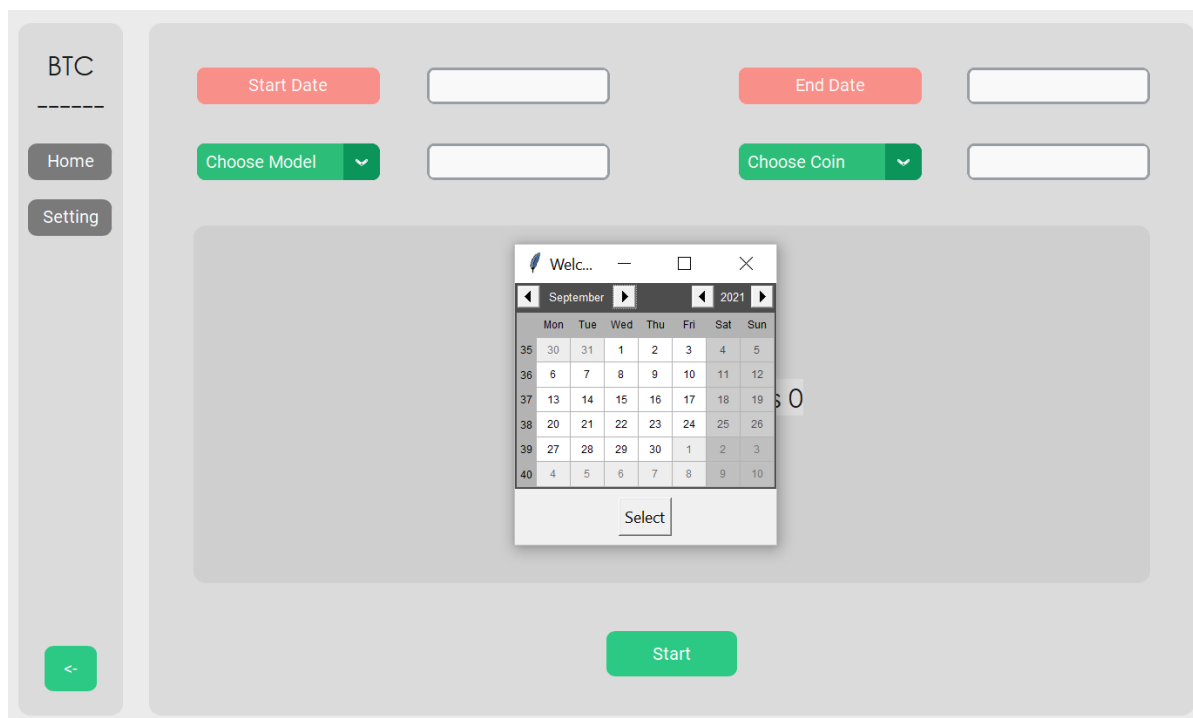


FIGURE 4.2 – Fenêtre de sélection des Dates

Le bouton "Start" lance la prédiction et affiche la date sélectionnée à partir du bouton "Date de fin", ainsi que la valeur prédite et la valeur réelle de clôture (close).

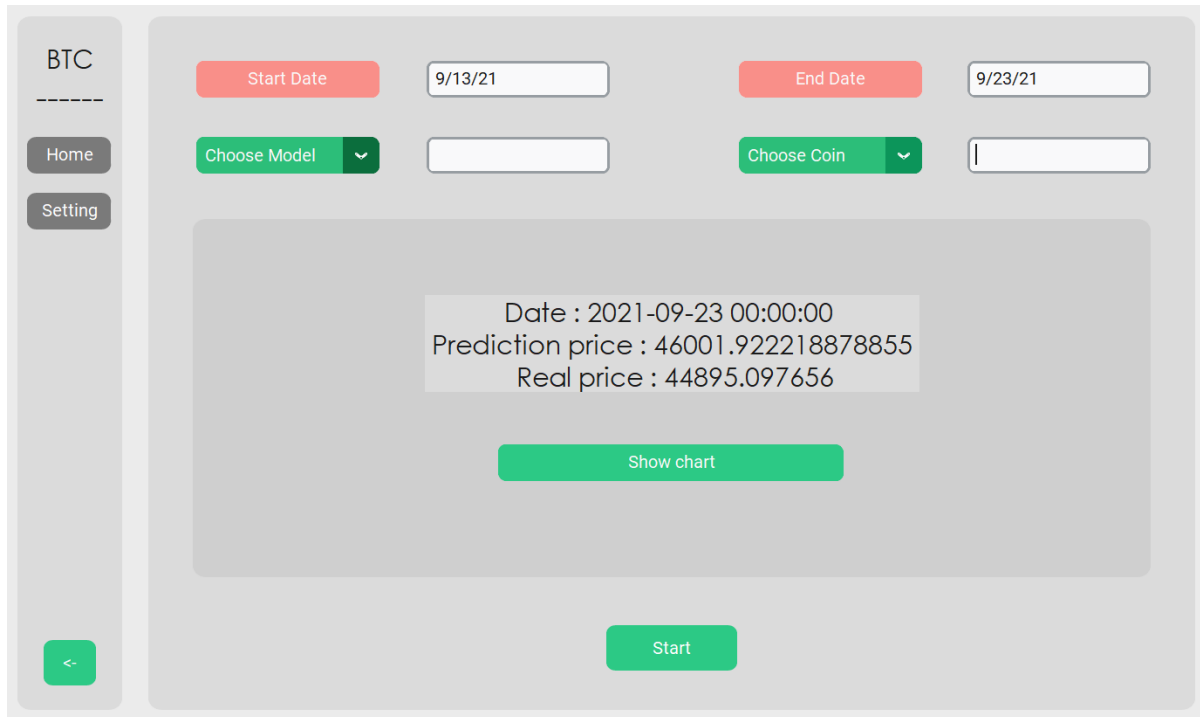


FIGURE 4.3 – Résultats de la prédiction

Le bouton "Show Chart" est le quatrième bouton dont nous avons parlé précédemment.



FIGURE 4.4 – Visualiser le graphique

4.3 Résultats et discussion

Nous présentons dans cette section les résultats que nous avons obtenus suite à notre travail. Nous commençons d'abord par la présentation des mesures de performances utilisées pour estimer la précision de notre prédiction.

4.3.1 Métriques de performance

- Mean Absolute Error (MAE) : est une mesure de performance calculée comme la moyenne des différences absolues entre les valeurs réelles et les valeurs prédites par le modèle de régression. Les valeurs de MAE sont comprises dans l'intervalle $(0, +\infty)$, et plus la valeur de MAE est basse, meilleure est la capacité de prédiction du modèle.

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^N |\hat{y}_i - y_i| \quad (4.1)$$

- Mean Squared Error (MSE) : est une mesure couramment utilisée pour évaluer les performances des modèles de régression. Elle quantifie la moyenne des différences au carré entre les valeurs réelles et prédites. Les valeurs de MSE peuvent varier de 0 à l'infini. Une valeur de MSE plus basse, se rapprochant de 0, indique un niveau de performance plus élevé du modèle, ce qui suggère que les prédictions sont plus proches des valeurs réelles.

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^N (\hat{y}_i - y_i)^2 \quad (4.2)$$

- Root Mean Squared Error (RMSE) : est une mesure de précision utilisée pour comparer les erreurs de prévision de différents modèles appliqués à un ensemble de données donné. Le RMSE représente la racine carrée des différences entre les valeurs prédites et les valeurs observées, ou la moyenne quadratique de ces différences. Une valeur plus basse de RMSE indique toujours de

meilleures performances du modèle.

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (\hat{y}_i - y_i)^2} \quad (4.3)$$

- Mean Absolute Percentage Error (MAPE) : est un indicateur largement utilisé pour évaluer la précision des prévisions ou des estimations. Il est calculé en prenant la moyenne des erreurs de pourcentage absolues entre les valeurs prédites et les valeurs réelles. Le MAPE peut varier de 0 à l'infini, et une valeur plus faible de MAPE indique une meilleure précision du modèle. Toutefois, il est important de noter que le MAPE peut être influencé par des valeurs nulles ou proches de zéro, ce qui peut rendre son interprétation moins fiable dans de tels cas. Par conséquent, il est recommandé d'utiliser d'autres métriques d'erreur lorsque les valeurs réelles peuvent être nulles ou très proches de zéro.

$$\text{MAPE} = \frac{1}{N} \sum_{i=1}^N \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100 \quad (4.4)$$

- \hat{y}_i est la valeur prédite, y_i est la valeur réelle et N est le nombre d'observations [61].

- Mean absolute relative error (MARE) : est une mesure statistique utilisée pour évaluer la précision des prédictions. Elle quantifie la déviation relative moyenne entre les valeurs prédites ou simulées et les valeurs observées correspondantes. Le MARE est exprimé comme une valeur non négative, et un MARE plus faible indique une meilleure précision.

$$\text{MARE} = \frac{1}{n} \sum_{i=1}^n \left| \frac{x_p - x_0}{x_0} \right| \quad (4.5)$$

- x_p : prédiction value , x_0 : actuel value , n est le nombre de cas de prédiction [62].

4.3.2 Résultats

Une fois le modèle entraîné, nous avons utilisé les données de test pour obtenir des prédictions, puis nous avons comparé les résultats avec les données réelles en utilisant des mesures de performance. Pour la validation de notre modèle, nous avons utilisé plusieurs indicateurs de performance tels que le MAE et le MSE.

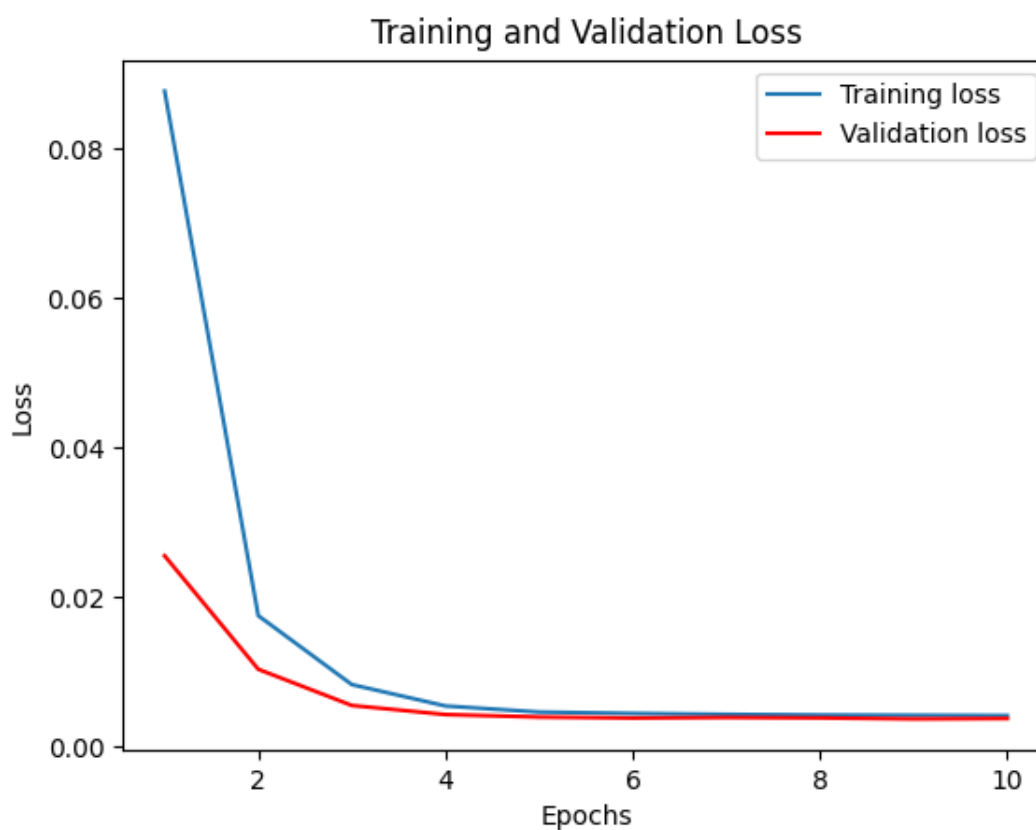
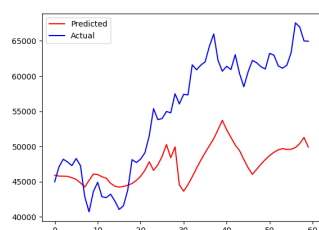


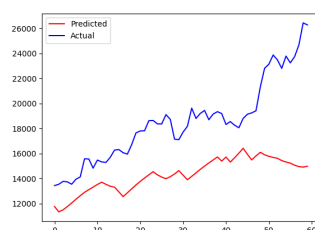
FIGURE 4.5 – Le Taux d’erreur (loss) de training et validation

Métrique	Valeur	val_Métrique	Valeur
MSE	0.0040	MSE	0.0038
MAE	0.0413	MAE	0.0438

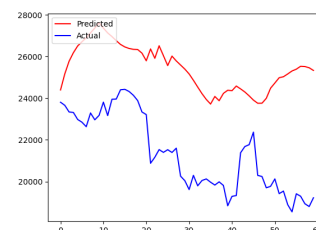
TABLE 4.1 – Le Taux d’erreur (loss) de training et validation



(A) La prédiction commence à partir du premier élément des données de test (Test_size = 0.2)



(B) Le modèle effectue des prédictions pendant une série de la période d’entraînement (Test_size = 0.3)



(C) Le modèle commence à faire des prédictions après les éléments de test initiaux (Test_size = 0.1)

FIGURE 4.6 – Comparaison des résultats de prédiction à partir de différentes zones de départ

Pour évaluer l’efficacité du modèle, nous avons expérimenté en modifiant le point de départ des prédictions, en commençant à partir des données de test et en initiant à partir de différentes régions. Étant donné que nous utilisons les résultats des prédictions comme entrées pour les prédictions ultérieures, il est très probable que l’erreur de prédiction augmente avec le temps.

Dans la Figure 4.6, nous montrons trois cas de test, dans la Figure 4.6a nous commençons la prédiction à partir d’une journée appartenant à la base d’apprentissage, notez que l’ensemble de prédictions (sauf pour la première) n’est pas réalisé sur des données déjà connues pour le modèle car les nouvelles valeurs de RSI changent à chaque nouvelle prédiction si la dernière est différente de la valeur réelle. La figure 4.6b commence à partir du point de séparation entre les données d’apprentissage et les données de test. Les mesures de performance pour ce cas sont données dans le tableau 4.2. Finalement, la Figure 4.6c montre les résultats de

prédiction à partir de données qui n'ont pas été vues par le modèle durant l'apprentissage. nous pouvons voir visuellement que notre modèle fait des prédictions de prix qui suivent la même tendance que les prix historiques du Bitcoin sans s'éloigner en terme de valeur absolue. Nous nous limitons à deux mois car pour des durées plus longues, on utilise des données hebdomadaire ou mensuelle au lieu des données quotidiennes.

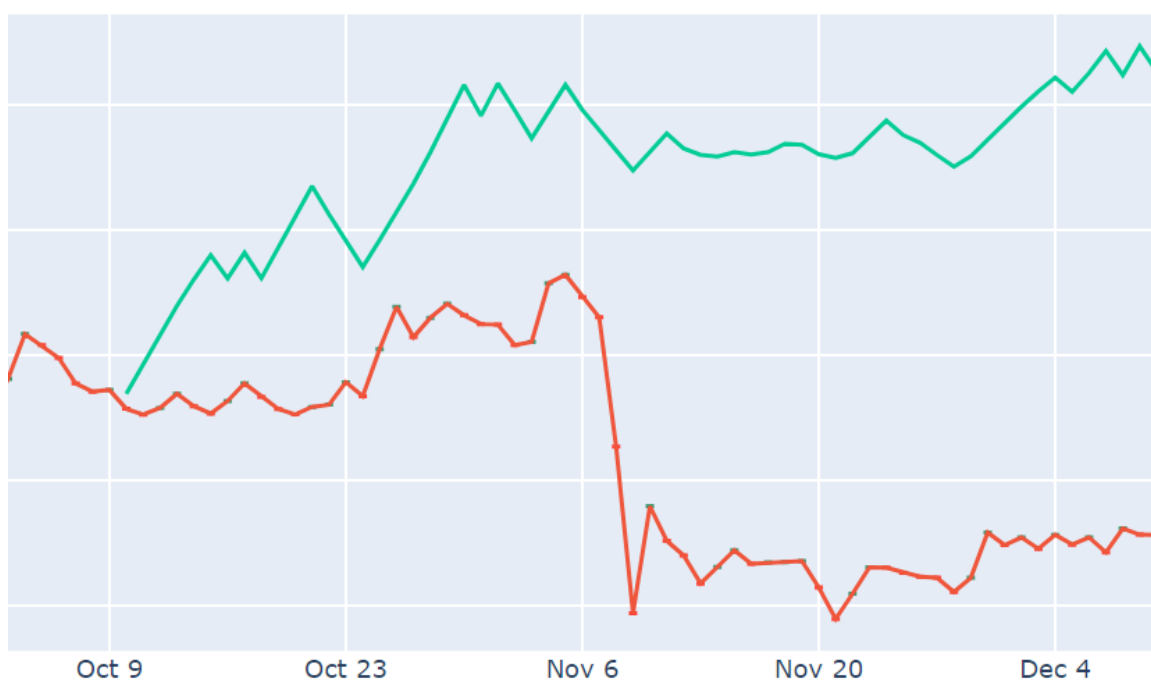


FIGURE 4.7 – Situation imprédictible - Faillite de FTX

Dans certains cas exceptionnels, la prédiction de notre modèle donne des valeurs qui sont très loin des valeurs réelles, cela est dû généralement à des événements politiques, des nouvelles lois ou un changement brutal dans le marché financier. Nous montrons comme exemple dans la Figure 4.7 où nous avons commencé la prédiction quelques jours avant la faillite de FTX, l'une des plus grande plateforme d'échange des crypto à l'époque. La valeur du Bitcoin a chuté de plus que 20% suite à cet événement qui est malheureusement impossible à prédire et modéliser, ce qui est encore une fois que tout modèle de prédiction financière, y compris le notre, ont des limites lorsqu'il s'agit d'événements similaires.

- Après avoir terminé la prédiction, nous évaluons le modèle sur plusieurs périodes, voici les résultats obtenus :

Durée de prévision	MARE	RMSE	MAPE
1 jour	0.02	0.014	2.11
7 jours	0.04	0.030	4.17
30 jours	0.07	0.063	7.04
60 jours	0.14	0.146	14.09

TABLE 4.2 – Résultats obtenus

4.4 Conclusion

Ce chapitre a été divisé en deux parties : l'implémentation et les résultats. Dans la première partie, nous avons présenté l'environnement de travail, le langage de programmation utilisé, les ensembles de données d'apprentissage et de test, ainsi que les détails de l'implémentation de la phase d'apprentissage et de l'interface graphique. La deuxième partie est consacrée à la présentation des résultats obtenus.

Nous avons également exploré d'autres modèles tels que LSTM et GRU, cependant, ils n'ont pas donné des résultats entièrement satisfaisants en utilisant les caractéristiques sur lesquelles nous avons travaillé. Cependant, le modèle MLP a donné des résultats relativement acceptables, qui pourraient être améliorés à l'avenir en prenant en compte d'autres indicateurs techniques.

CONCLUSION GÉNÉRALE

Dans ce travail, nous avons exploré le monde des cryptomonnaies, en mettant l'accent particulier sur Bitcoin. Nous avons discuté de sa création et de la manière dont il a gagné une popularité significative, ainsi que des raisons qui ont suscité un intérêt considérable. Nous avons étudié les dernières recherches sur la prédiction du prix de Bitcoin et d'autres cryptomonnaies. Nous avons également mentionné certains indicateurs qui aident à la prévision.

Nous avons proposé un modèle de prédiction du prix de Bitcoin, à savoir le MLP (Multi-Layer Perceptron). Le choix du MLP a été motivé par sa capacité à capturer les relations non linéaires, à extraire les caractéristiques pertinentes et à s'adapter aux conditions changeantes du marché, ce qui en fait une approche adaptée à la prédiction des cryptomonnaies. Il a donné de meilleurs résultats que les autres modèles que nous avons essayés, tels que le LSTM et le GRU.

Nous avons également utilisé dans ce travail l'indicateur RSI (Relative Strength Index), qui est un indicateur important qui nous a aidé dans le processus de prédiction. Pour évaluer notre modèle, nous avons utilisé des métriques de performance bien connues. Nous avons obtenu des résultats relativement acceptables qui peuvent être améliorés à l'avenir. La prédiction des devises numériques est difficile en raison de leur volatilité et de leur sensibilité à divers facteurs. Cependant, ce travail représente une bonne première étape vers l'obtention de meilleurs résultats en utilisant

une approche basée sur les indicateurs techniques.

Plusieurs autres indicateurs techniques ou facteurs fondamentaux peuvent être estimés et utilisés afin de donner à notre modèle plus d'informations sur l'historique du marché et lui permettre de prédire avec plus de précision les futures valeurs. Néanmoins, le choix des meilleurs facteurs est une tâche combinatoire qui peut prendre beaucoup de temps à tester et valider. Une avenue de recherche supplémentaire pouvant être explorée pour améliorer encore davantage la précision et la robustesse du modèle est de développer un mécanisme automatique de sélection des indicateurs techniques les plus pertinents pour la prédiction du prix du Bitcoin. Cela pourrait être réalisé en utilisant des techniques d'apprentissage automatique supplémentaires, telles que l'apprentissage par renforcement, pour identifier les indicateurs qui ont la plus grande influence sur la performance du modèle.

BIBLIOGRAPHIE

- [1] Mariette AWAD et Rahul KHANNA. « Machine Learning ». In : *Efficient Learning Machines : Theories, Concepts, and Applications for Engineers and System Designers*. Berkeley, CA : Apress, 2015, p. 1-18. ISBN : 978-1-4302-5990-9. DOI : 10 . 1007 / 978-1-4302-5990-9_1. URL : https://doi.org/10.1007/978-1-4302-5990-9_1.
- [2] Rami AHMAD, Raniyah WAZIRALI et Tarik ABU-AIN. « Machine learning for wireless sensor networks security : An overview of challenges and issues ». In : *Sensors* 22.13 (2022), p. 4730.
- [3] Alex SMOLA et S.V.N. VISHWANATHAN. *Introduction to Machine Learning*. 2008. ISBN : 0 521 82583 0.
- [4] Wendbenedo ARNAUD. « Application des algorithmes d'apprentissage automatique pour la détection de défauts de roulements sur les machines tournantes dans le cadre de l'Industrie 4.0 ». Thèse de doct. Université du Québec à Chicoutimi, 2020.
- [5] Sun Y. V. König I. R. Bailey-Wilson J. E. Malley J. D. DASGUPTA A. « Brief review of regression-based and machine learning methods in genetic epidemiology : the Genetic Analysis Workshop 17 experience ». In : (2011). DOI : <https://doi.org/10.1002/gepi.20642>.

- [6] Salim DRIDI. « Supervised Learning - A Systematic Literature Review ». In : (2022). DOI : 10.31219/osf.io/tysr4.
- [7] Hongyu LIU et Bo LANG. « Machine Learning and Deep Learning Methods for Intrusion Detection Systems : A Survey ». In : *Applied Sciences* 9.20 (2019). ISSN : 2076-3417. DOI : 10.3390/app9204396. URL : <https://www.mdpi.com/2076-3417/9/20/4396>.
- [8] Christian JANIESCH, Patrick ZSCHECH et Kai HEINRICH. « Machine learning and deep learning ». In : *Electronic Markets* 31.3 (2021), p. 685-695.
- [9] Rishabh CHOUDHARY et Hemant GIANEY. « Comprehensive Review On Supervised Machine Learning Algorithms ». In : déc. 2017, p. 37-43. DOI : 10.1109/MLDS.2017.11.
- [10] Awodele O. Hinmikaiye J. O.-Olakanmi O. Akinjobi J. OSISANWO F.Y. Akinsola J.E.T. « Supervised Machine Learning Algorithms : Classification and Comparison ». In : *International Journal of Computer Trends and Technology (IJCTT)* 33.3 (2017). DOI : 10.14445/22312803/IJCTT-V48P126.
- [11] J Ross QUINLAN. « Learning decision tree classifiers ». In : *ACM Computing Surveys (CSUR)* 28.1 ((1996)), p. 71-72.
- [12] Liu YINGQIU, Li WEI et Lin YUNCHUN. « Network Traffic Classification Using K-means Clustering ». In : *Second International Multi-Symposiums on Computer and Computational Sciences (IMSCCS 2007)* (2007), p. 360-365.
- [13] Gopi BATTINENI, Nalini CHINTALAPUDI et Francesco AMENTA. « Machine learning in medicine : Performance calculation of dementia prediction by support vector machines (SVM) ». In : *Informatics in Medicine Unlocked* 16 (2019), p. 100200. ISSN : 2352-9148. DOI : <https://doi.org/10.1016/j.imu.2019.100200>. URL : <https://www.sciencedirect.com/science/article/pii/S2352914819300917>.
- [14] K. Ming LEUNG. « Naive Bayesian Classifier ». In : (2007).

- [15] Jafar Ahmad Abed ALZUBI, Anand NAYYAR et Akshi KUMAR. « Machine Learning from Theory to Algorithms : An Overview ». In : *Journal of Physics : Conference Series* 1142 (2018).
- [16] Zanifa OMARY et Fredrick MTENZI. « Machine learning approach to identifying the dataset threshold for the performance estimators in supervised learning ». In : *ACM Computing Surveys (CSUR)* 3.3 (2010), p. 314-325.
- [17] Alaa THARWAT. « Linear vs. quadratic discriminant analysis classifier : a tutorial ». In : *International Journal of Applied Pattern Recognition* 3.2 (2016), p. 145-180.
- [18] Iqbal H SARKER. « Machine learning : Algorithms, real-world applications and research directions ». In : *SN computer science* 2.3 (2021), p. 160.
- [19] Keiron O'SHEA et Ryan NASH. « An Introduction to Convolutional Neural Networks ». In : *CoRR* abs/1511.08458 (2015). arXiv : 1511 . 08458. URL : <http://arxiv.org/abs/1511.08458>.
- [20] Xiaozhou QIN et al. « A Combined Gated Recurrent Unit and Multi-Layer Perception Neural Network Model for Predicting Shale Gas Production ». In : *Processes* 11.3 (2023). ISSN : 2227-9717. DOI : 10 . 3390 /pr11030806. URL : <https://www.mdpi.com/2227-9717/11/3/806>.
- [21] Aref HASHEMI FATH, Farshid MADANIFAR et Masood ABBASI. « Implementation of multilayer perceptron (MLP) and radial basis function (RBF) neural networks to predict solution gas-oil ratio of crude oil systems ». In : *Petroleum* 6.1 (2020), p. 80-91. ISSN : 2405-6561. DOI : <https://doi.org/10.1016/j.petlm.2018.12.002>. URL : <https://www.sciencedirect.com/science/article/pii/S2405656118301020>.
- [22] Eduardo TODT et Bruno Alexandre KRINSKI. « Introduction to the Artificial Neural Networks ». In : (2011). DOI : <https://doi.org/10.5772/15751>.
- [23] A. BORS. « Introduction Of The Radial Basis Function Rbf Networks ». In : 2000.

- [24] Krenker ANDREJ, Bester JANEZ et Kos ANDREJ. « Introduction to the Artificial Neural Networks ». In : (2011). DOI : <https://doi.org/10.5772/15751>.
- [25] Mobarak ABUMOHSEN, Amani Yousef OWDA et Majdi OWDA. « Electrical load forecasting using LSTM, GRU, and RNN algorithms ». In : *Energies* 16.5 (2023), p. 2283.
- [26] Mohammad J HAMAYEL et Amani Yousef OWDA. « A novel cryptocurrency price prediction model using GRU, LSTM and bi-LSTM machine learning algorithms ». In : *AI* 2.4 (2021), p. 477-496.
- [27] Xuan-Hien LE et al. « Application of Long Short-Term Memory (LSTM) Neural Network for Flood Forecasting ». In : *Water* 11.7 (2019). ISSN : 2073-4441. DOI : [10.3390/w11071387](https://doi.org/10.3390/w11071387). URL : <https://www.mdpi.com/2073-4441/11/7/1387>.
- [28] Phumudzo Lloyd SEABE, Claude Rodrigue Bambe MOUTSINGA et Edson PINDZA. « Forecasting Cryptocurrency Prices Using LSTM, GRU, and Bi-Directional LSTM: A Deep Learning Approach ». In : *Fractal and Fractional* 7.2 (2023). ISSN : 2504-3110. DOI : [10.3390/fractalfract7020203](https://doi.org/10.3390/fractalfract7020203). URL : <https://www.mdpi.com/2504-3110/7/2/203>.
- [29] Songjune LEE, Seungjin KANG et Gwang-Se LEE. « Predictions for Bending Strain at the Tower Bottom of Offshore Wind Turbine Based on the LSTM Model ». In : *Energies* 16.13 (2023), p. 4922.
- [30] Krzysztof ZARZYCKI et Maciej ŁAWRYŃCZUK. « LSTM and GRU Neural Networks as Models of Dynamical Processes Used in Predictive Control : A Comparison of Models Developed for Two Chemical Reactors ». In : *Sensors* 21.16 (2021). ISSN : 1424-8220. DOI : [10.3390/s21165625](https://doi.org/10.3390/s21165625). URL : <https://www.mdpi.com/1424-8220/21/16/5625>.
- [31] Ingolf Gunnar Anton PERNICE et Brett SCOTT. « Cryptocurrency ». In : *Internet Policy Review, Glossary of decentralised technosocial systems* 10.2 (2021).
- [32] Krishna Kumar THAKUR et Gour Gopal BANIK. « Cryptocurrency : Its Risks And Gains And The Way Ahead ». In : 2018.

- [33] Fan FANG et al. « Cryptocurrency trading : a comprehensive survey ». In : *Financial Innovation* 8.1 (2022), p. 1-59.
- [34] Fang FAN et al. « Cryptocurrency trading : a comprehensive survey ». In : *Financial Innovation* 8 (déc. 2022). DOI : 10.1186/s40854-021-00321-6.
- [35] O AL MASHHOUR et A ABD AZIZ. « The Era of Cryptocurrencies : A Study About the Advantages and Disadvantages ». In : *The Proceeding of The First Sois Conference on Arts and Humanities (Sicah) 2019 reshaping sustainable development agenda through arts & humanities*. 2019, p. 10-20.
- [36] <https://coinmarketcap.com/alexandria/glossary/trading-volume>.
- [37] <https://academy.binance.com/fr/articles/crypto-market-capitalization-explained>.
- [38] <https://www.investopedia.com/terms/m/movingaverage.asp>.
- [39] <https://www.investopedia.com/trading/support-and-resistance-basics/>.
- [40] Murtadha ALHILFI et al. « Role of using the Relative Strength Index in Making Speculation Decision in Stock Applied Research in the Iraq Stock Exchange ». In : *Int. J. Acad. Res. Accounting, Financ. Manag. Sci* 9 (2019), p. 123-135.
- [41] <https://www.coinbase.com/fr/learn/crypto-basics/what-is-volatility>.
- [42] <https://www.investopedia.com/terms/c/cryptocurrency.asp>.
- [43] <https://www.investopedia.com/terms/s/speculation.asp>.
- [44] <https://www.binance.com/en/blog/fiat/what-is-market-manipulation-in-cryptocurrency-421499824684902912>.
- [45] Mohammad J. HAMAYEL et Amani Yousef OWDA. « A Novel Cryptocurrency Price Prediction Model Using GRU, LSTM and bi-LSTM Machine Learning Algorithms ». In : *AI* 2.4 (2021), p. 477-496. ISSN : 2673-2688. DOI : 10.3390/ai2040030.

- [46] Muhammad RIZWAN, Sanam NAREJO et Moazzam JAVED. « Bitcoin price prediction using Deep Learning Algorithm ». In : *2019 13th International Conference on Mathematics, Actuarial Science, Computer Science and Statistics (MACS)*. 2019, p. 1-7. DOI : 10.1109/MACS48846.2019.9024772.
- [47] Alvin HO, R VATAMBETI et SK RAVICHANDRAN. « Bitcoin Price Prediction Using Machine Learning and Artificial Neural Network Model ». In : *Indian Journal of Science and Technology* 14.27 (2021), p. 2300-2308.
- [48] Abdullah H. AL-NEFAIE et Theyazn H. H. ALDHYANI. « Bitcoin Price Forecasting and Trading : Data Analytics Approaches ». In : *Electronics* 11.24 (2022). ISSN : 2079-9292. DOI : 10.3390/electronics11244088. URL : <https://www.mdpi.com/2079-9292/11/24/4088>.
- [49] Phumudzo Lloyd SEABE, Claude Rodrigue Bambe MOUTSINGA et Edson PINDZA. « Forecasting Cryptocurrency Prices Using LSTM, GRU, and Bi-Directional LSTM : A Deep Learning Approach ». In : *Fractal and Fractional* 7.2 (2023). ISSN : 2504-3110. DOI : 10.3390/fractalfract7020203. URL : <https://www.mdpi.com/2504-3110/7/2/203>.
- [50] Svend PASAK et Riyanto JAYADI. « Investment Decision on Cryptocurrency : Comparing Prediction Performance Using ARIMA and LSTM ». In : *Journal of Information Systems and Informatics* 5.2 (2023), p. 407-427.
- [51] V¹ AKILA et al. « A Cryptocurrency Price Prediction Model using Deep Learning ». In : *E3S Web of Conferences*. T. 391. EDP Sciences. 2023, p. 01112.
- [52] Rifando PANGGABEAN et Yohana Dewi Lulu WIDYASARI. « A comparison between Super Vector Regression, Random Forest Regressor, LSTM, and GRU in Forecasting Bitcoin Price ». In : *International ABEC* (2023), p. 281-287.
- [53] Nor Azizah HITAM et Amelia Ritahani ISMAIL. « Comparative Performance of Machine Learning Algorithms for Cryptocurrency Forecasting ». In : *Indonesian Journal of Electrical Engineering and Computer Science* (2018).
- [54] <https://docs.python.org/3/tutorial>.

-
- [55] <https://www.tensorflow.org/overview>.
- [56] <https://www.tensorflow.org/guide/keras?hl=fr>.
- [57] https://scikit-learn.org/stable/getting_started.html?hl=fr.
- [58] <https://numpy.org/doc/stable/>.
- [59] <https://pandas.pydata.org>.
- [60] <https://finance.yahoo.com/quote/BTC-USD/history/>.
- [61] Makrufa Sh HAJIRAHIMOVA et Aybeniz S ALIYEVA. « Development of a Prediction Model on Demographic Indicators based on Machine Learning Methods : Azerbaijan Example ». In : (2023).
- [62] Thomas P OSCAR. « Response surface models for effects of temperature, pH, and previous growth pH on growth kinetics of Salmonella Typhimurium in brain heart infusion broth ». In : *Journal of Food Protection* 62.2 (1999), p. 106-111.