الجمهورية الجزائرية الديمقراطية الشعبية
REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
وزارة التعليم العالي و البحث العلمي
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
– جامعة أبي بكر بلقايد –تلمسان
Université Aboubakr Belkaïd – Tlemcen –
Faculté de TECHNOLOGIE

MEMOIRE

Présenté pour l'obtention du **diplôme** de **MASTER**

Réaliser par : BOUDGHENE STAMBOULI Ali

HABRI Mohamed Nazim

Spécialité : Télécommunications

Filière : Réseau et Télécommunications

Intégration de l'intelligence artificielle pour la sécurité des réseaux IoT

Soutenu publiquement, le 29/06/2025, devant le jury composé de :

Mme. MELIANI MAGHNIA	Professeur	Université de Tlemcen	Président
Mme. BOUCHNAK KHELADI YASMINA	MCB	Université de Tlemcen	Examinateur
Mme. BENAISSA AMAL	MCB	Université de Tlemcen	Encadrant
Mme.TALEB Sarra	MCB	Université de Tlemcen	Co-encadrant

Année Universitaire : 2024/2025

Remerciements

Nous tenons à exprimer notre reconnaissance envers ALLAH, source de sagesse et de guidance. Nous avons foi en sa bonté qui a éclairé nos voies à chaque tournant de ce parcours académique.

Nous tenons à exprimer notre profonde gratitude envers toutes les personnes qui ont contribué, de près ou de loin, à la réalisation de ce projet de fin d'études.

En premier lieu, nous souhaitons adresser nos sincères remerciements à nos encadrants, pour leur guidance précieuse, leurs conseils avisés et leur disponibilité tout au long de ce travail. Leur patience a été essentielle à l'aboutissement de cette recherche.

Nous remercions également les membres du jury, d'avoir accepté d'évaluer ce travail. Leurs remarques et suggestions seront très utiles pour la suite.

Nous souhaitons également exprimer notre reconnaissance à la bibliothèque Mohamed DIB. Leur accès aux ressources a été grandement apprécié.

Enfin, nous tenons à remercier chaleureusement nos familles et nos amis pour leur soutien constant, leurs encouragements et leur patience durant cette période intense de travail.

Ce projet est le fruit de tous ces efforts combinés, et nous leur en sommes profondément reconnaissants

Résumé

À l'échelle mondiale, des milliards d'objets autonomes et intelligents établissent une communication entre eux tout en étant interconnectés. Effectivement, l'Internet des Objets occupe aujourd'hui une part importante de notre vie quotidienne. Ce concept révolutionnaire a crée une nouvelle dimension qui efface les frontières entre le monde numérique et le monde physique. Ce système connaît une succès considérable grâce aux technologies de communication, notamment les technologies sans fil, ainsi qu'aux dispositifs matériels. L'IoT couvre presque tous les secteurs actuels de la technologie de l'information (IT) et est le résultat de l'intégration de diverses technologies et de leur évolution. Le fait que les appareils de l'IoT puissent analyser, communiquer, gérer et traiter des données de manière automatique et sans intervention humaine attribue à l'IoT une puissance considérable.

Dans ce travail de recherche, notre objectif est de concevoir un système de sécurité capable d'assurer la surveillance et l'identification d'attaques dans un environnement IoT. Nous avons constaté une absence significative de solutions capables d'examiner et de repérer différentes sortes d'attaques au sein des réseaux IoT. Pour résoudre ce problème, nous avons mis au point des systèmes de détection d'intrusion comportementale basés sur l'apprentissage automatique et l'apprentissage profond, qui permettent d'établir un modèle de communications authentiques. Il faut tenir compte des limitations des objets et des technologies de communication employés dans ces méthodes.

Mot clés : *Internet des objets (IoT), attaques, détection d'intrusion,* cybersécurité, *apprentissage automatique, réseaux de neurones, apprentissage profond, modèles prédictifs.*

Abstract

Globally, billions of autonomous and intelligent objects communicate with each other while being interconnected. Indeed, the Internet of Things (IoT) now occupies a significant part of our daily lives. This revolutionary prototype has created a new dimension that erases the boundaries between the digital and physical worlds. This product is enjoying considerable success thanks to communication technologies, particularly wireless, as well as hardware devices. The IoT covers almost all current sectors of information technology (IT) and is the result of the integration and evolution of various technologies. The fact that IoT devices can analyze, communicate, manage, and process data automatically and without human intervention gives IoT considerable power.

In this research, our objective is to create a security system capable of monitoring and identifying attacks in an IoT context. We noted a significant lack of solutions capable of examining and detecting different types of attacks within an IoT network. To address this problem, we have developed behavioral intrusion detection systems based on machine learning and deep learning, which enable us to model authentic communications.

The limitations of the devices and communication technologies used in these methods must be taken into account.

Keywords: *Internet of Things (IoT), anomalies, intrusion detection, cybersecurity, machine learning, neural networks, deep learning, predictive models.*

ملخص

على الصعيد العالمي ، تتواصل مليارات الأجهزة الذكية والمستقلة مع بعضها البعض مترابطة في آنٍ واحد وفي الواقع، أصبح إنترنت الأشياء يشغل الآن جزءًا كبيرًا من من حياتنا اليومية وقد أنشأ هذا الهيكل الثوري بُعدًا جديدًا يلغي الحدود بين العالمين الرقمي والمادي .ويشهد هذا النظام نجاحًا كبيرًا بفضل تقنيات الاتصال، وخاصةً التقنيات اللاسلكية، بالإضافة إلى الأجهزة المادية يغطي إنترنت الأشياء جميع قطاعات تكنولوجيا المعلومات الحالية تقريبًا، وهو ثمرة تكامل وتطور مختلف التقنيات ،إن قدرة أجهزة إنترنت الأشياء على تحليل البيانات تتوقعها وإدارتها ومعالجتها وبالتالي تؤثر على قوة بشرية هامة.

في هذا البحث، نهدف إلى تصميم نظام أمان قادر على رصد وتحديد الهجمات في بيئة إنترنت الأشياء .وقد لاحظنا نقصًا كبيرًا في الحلول القادرة على فحص وكشف أنواع مختلفة من الهجمات داخل شبكات إنترنت الأشياء .ولمعالجة هذه المشكلة، طورنا أنظمة كشف سلوكية عن التسلل تعتمد على التعلم الآلي والتعلم العميق، مما يُمكّننا من نمذجة اتصالات حقيقية .يجب مراعاة قيود الأجهزة وتقنيات الاتصال المستخدمة في هذه الأساليب.

الكلمات المفتاحية :الهجمات، كشف التسلل، الأمن السيبراني ،(IoT) إنترنت الأشياء والآلي، الشبكات العصبية، التعلم العميق، النماذج التنبؤية

TABLE DE MATIERE

LISTE DE FIGURES

LISTE DES TABLEAUX

Liste des Abréviations

IoT Internet of Things

IPv4 Internet Protocol version 4

IPv6 Internet Protocol version 6

M2M Machine to Machine

PAN Personal area network

NFC Near Field Communication

RF Random forest

IT information technology

RFID Radio-Frequency Identification

ONS Objet name service

ABE Attribute Based Encryption

DDOS Distributed Denial of Service

DOS denial of service

HTTP Hypertext Transfer Protocol

TCP Transmission control protocol

SDN Software-Defined Networking

ACK acknowledgment of receipt

ICMP Internet control message protocol

UDP User datagram protocole

MQTT Message Queuing Telemetry Transport

CoAP Constrained Application Protocol

6loWPAN Low Power Wireless Area Networks.

AES Advanced Encryption Standar

CBC Cipher block chaining

RSA Rivest, Shamir et Adleman

DSA Data structure algorithme

IBE Identity Based Encryption

IDS Intrusion Detection System

NIDS Network Intrusion Detection System

DTLS datagram transport layer security

ML Machine learning

DL Deep learning

DMZ demilitarized zone

ANN Artificial Neural Network

SVM supervised machine learning algorithm

K-NN K-Nearest neighbors

RNN Recurrent neural networks

LSTM Long Short-Term Memory

MLP Multi-Layer Perceptron

WSN Wireless Sensor Network

Introduction générale

Aujourd'hui, les progrès technologiques constituent sans conteste la base sur laquelle s'appuient la majorité des entreprises à travers le monde. Effectivement, les responsables de grandes villes et les collectivités dans de nombreux pays, principalement les plus avancés, envisagent sérieusement la mise en place ou la création de smart cities. L'objectif est d'offrir à leurs résidents des services intelligents qui amélioreront la qualité de vie humaine en accordant une attention particulière à l'aspect écologique. Il est désormais possible de concevoir un monde intelligent qui interagit avec le développement durable, protège la vie privée des citoyens et garantit une meilleure qualité de vie.

L'adoption du modèle de l'Internet des Objets établit un lien entre l'univers urbain et l'univers numérique, le premier étant connecté au second. C'est la là conception d'un monde intelligent. L'IoT propose une variété unique de services et d'applications en facilitant l'interconnexion de divers objets, machines et capteurs, qui peuvent être utilisés par des individus. L'objectif de la connexion de divers objets est d'automatiser certaines activités quotidiennes afin de simplifier l'existence humaine, grâce à l'utilisation de technologies récentes appropriées et des fonctionnalités proposées par l'informatique en cloud. L'IoT a ainsi une influence socio-économique significative, car il permet non seulement de diminuer les coûts d'instauration des services, mais aussi d'améliorer la qualité de vie.

Il est crucial de considérer la sécurité des données dans un contexte IoT afin d'assurer la protection des individus et de l'écosystème contre les conséquences d'actes dangereux. Cela inclut également la capacité à détecter et identifier les risques et menaces. L'IoT pose des enjeux significatifs en termes de sécurité des données.

Effectivement, les informations sensibles sont mises en danger et la sécurité personnelle est exposée aux cyberattaques lorsque les outils et services IoT insuffisamment sécurisés deviennent des points d'accès vulnérables. Par conséquent, la fourniture de services essentiels et publics peut être compromise, et la solidité de l'Internet dans son ensemble est potentiellement mise en risque, ce qui peut compromettre la protection de la vie privée et la sécurité individuelle. Cela pourrait mettre en danger les utilisateurs ou les infrastructures si des attaques étaient menées contre des réseaux et appareils IoT mal protégés. On peut ainsi former des réseaux de dispositifs contrôlés à distance et reliés à Internet, généralement appelés réseaux de robots, en exploitant, par exemple, des webcams ou des caméras de sécurité compromise en tant qu'appareils IoT.

Dans ce cadre, ces appareils sont appelés « botnets » et sont employés pour s'attaquer à l'infrastructure Internet, à d'autres utilisateurs ou réseaux, généralement après avoir été contaminés par un logiciel malveillant à des fins perturbatrices ou criminelles. Effectivement, c'est arrivé en 2016 quand les plateformes majeures comme Amazon, Netflix et Twitter ont connu une indisponibilité temporaire dans diverses régions du monde pour leurs utilisateurs. Cela était dû à une attaque par déni de service distribué (DDoS) visant un fournisseur majeur de services DNS, ciblé par un réseau de dispositifs IoT affecté.

Structure de Mémoire

Chapitre 1 : Ce chapitre présente un aperçu complet de l'Internet des Objets, à travers les définitions et architectures proposées par divers organismes de normalisation et normes internationales. Par ailleurs, une analyse comparative des différents protocoles et technologies de communication utilisés dans un contexte IoT

est effectuée. À la conclusion de ce chapitre, nous mettons en lumière les défis technique et les avantages d'un système IoT.

Chapitre 2 : est introduit par une brève présentation des mesures de sécurité essentielles dans l'Internet des Objets (IoT). Par la suite, nous présentons les différentes catégories de vulnérabilités ainsi que les types d'attaques qui en tirent parti. Ensuite, Nous exposons les concepts fondamentaux de la sécurité, tels que l'authentification, l'intégrité, la non-répudiation et la confidentialité, ainsi que la stratégie de sécurité relative à l'IoT. Nous proposons également un panorama des principales méthodes de sécurité suggérées dans les articles, qui visent à répondre aux enjeux de sécurité et à créer une règle efficace en vue de repérer et d'éradiquer les assauts les plus courants dans l'IoT. Pour conclure, ce chapitre offre un aperçu des recherches à venir dans ce domaine.

Chapitre 3 : Présente une revue de la littérature sur le contexte global de l'apprentissage automatique, en abordant les définitions et en expliquant le cadre d'application utilisé. En outre, nous décrivons les algorithmes de machine learning utilisés pour justifier le choix des modèles retenus dans notre étude , ainsi que les plateformes mises en œuvre. Par la suite, nous présentons un ensemble de techniques de détection d'anomalies reposant sur l'utilisation d'algorithmes de deep learning, ainsi que les critères d'évaluation adoptés pour en analyser les performances.

Chapitre 4 : Dans ce chapitre, nous exposons en détail nos contributions à la sécurisation des réseaux IoT face aux différents types d'attaques, en particulier dans un environnement caractérisé par l'hétérogénéité des dispositifs, les ressources limitées et la multiplicité des vecteurs de menaces. Deux propositions principales sont mises en avant :

- En premier lieu, nous proposons un Système de Détection d'Intrusion basé sur l'Apprentissage Automatique (ML-NIDS). Ce système repose sur des algorithmes classiques de machine learning permettant d'identifier aussi bien les attaques récurrentes que les anomalies peu fréquentes. Il se distingue par sa capacité à offrir un bon équilibre entre précision, vitesse de détection et réduction des faux positifs.
- En second lieu, nous développons un Système de Détection d'Intrusion basé sur l'Apprentissage Profond (DL-NIDS). Ce second système exploite des architectures de réseaux neuronaux profonds afin de modéliser les schémas complexes de comportement au sein des communications IoT. Il permet une détection fine et automatisée des menaces multiples, avec de meilleures performances notamment en classification multiclasses.

Les expérimentations menées sur la base de données CIC_IoT_23 démontrent que les deux approches proposées permettent d'atteindre des taux de détection élevés et une précision supérieure par rapport à plusieurs modèles de référence dans la littérature.

Objectifs

L'objectif principal de ce travail est d'assurer la sécurité des réseaux de l'Internet des Objets (IoT), face à la diversité des menaces et attaques auxquelles ils sont exposés. En effet, la plupart des systèmes de détection d'intrusion existants ne sont pas conçus pour évoluer dans des environnements dynamiques comme ceux des réseaux IoT, où les nœuds intègrent des capteurs, des actionneurs, une logique de traitement et des interfaces de communication.

De plus, les solutions traditionnelles se limitent souvent à un nombre restreint de scénarios d'attaque, ce qui les rend inadaptées face aux menaces émergentes et de plus en plus sophistiquées.

Ainsi, l'un des objectifs de ce mémoire est de développer un système de détection d'intrusion flexible et adaptatif aux réseaux IoT, capable de réagir efficacement à un large éventail d'intrusions.

Nous proposons ainsi dans le cadre de ce mémoire des approches basées sur l'intelligence artificielle. Ces solutions portent sur des systèmes de détection d'intrusion (IDS) intelligents, reposant sur deux approches majeures : l'apprentissage automatique (machine learning) et l'apprentissage profond (deep learning). Ces approches visent à garantir une détection efficace et une meilleure résilience face aux attaques, y compris les plus sophistiquées.

CHAPITRE 1

FONDEMENTS DE L'INTERNET DES OBJETS

1.1 INTRODUCTION

L'Internet des Objets fait référence à un ensemble d'objets physiques connectés à Internet, en mesure de recueillir, traiter et partager des informations sans l'implication directe d'un utilisateur. Ces dispositifs intelligents intègrent des capteurs, des logiciels et des technologies de communication pour interagir avec leur environnement et exécuter des décisions autonomes. Ce chapitre explore les origines, les principes fondamentaux et l'impact croissant de l'IoT dans divers domaines. Il illustre comment l'internet des objets transforme l'industrie, la santé, l'agriculture, la logistique et d'autres secteurs, en s'appuyant sur des exemples concrets.

1.2 FONCTIONNEMENT DE L'INTERNET DES OBJETS

1.2.1 Appareils intelligents : Ce sont des dispositifs, tel qu'une télévision, une caméra de surveillance ou un équipement de fitness, qui intègrent des fonctionnalités informatiques. Il acquiert des informations de son milieu, des interactions effectuées par l'utilisateur ou des schémas d'utilisation, puis les transmettent via Internet, vers et depuis leur application IoT.

1.2.2 Application pour l'Internet des Objets : Une application IoT est un ensemble de services et de logiciels qui collecte les informations envoyées par différents dispositifs connectés. Elle utilise des technologies d'intelligence artificielle afin d'examiner ces informations et de prendre des décisions pertinentes. Ces décisions sont ensuite transmises à l'équipement IoT, qui réagit de manière intelligente.

1.2.3 Interface graphique pour l'utilisateur :

La gestion d'un appareil IoT ou d'une flotte d'appareils peut être effectuée via une interface graphique. Un exemple courant est une application mobile ou un site web permettant de configurer, contrôler et superviser des dispositifs connectés.

1.3 CONCEPTS FONDAMENTAUX

Dans le paysage foisonnant des technologies sans fil, un segment crucial est occupé par les solutions à portée intermédiaire. Ces technologies, à l'instar du Wi-Fi, du Zigbee et du Bluetooth Low Energy (BLE), jouent un rôle essentiel dans une multitude d'applications, allant de la connectivité domestique et personnelle aux réseaux de capteurs industriels. Elles se situent entre les technologies à courte portée, comme le NFC, et les technologies à longue portée, telles que les réseaux cellulaires. Chacune possède ses propres caractéristiques en termes de débit de données, de consommation d'énergie, de portée et de topologie de réseau, les rendant adaptées à des cas d'usage spécifiques. Cette introduction explorera brièvement les fondements et les applications clés de ces trois technologies omniprésentes.

1.4 TECHNOLOGIES A PORTEE INTERMEDIAIRE

1.4.1 WiFi

Il constitue un réseau sans fil localisé, caractérisé par une consommation d'énergie importante, qui répond exclusivement aux appareils connectés au réseau électrique ou à ceux qui sont alimentés de manière simple et régulière. Ce réseau permet une transmission rapide de grandes quantités de données et est compatible avec les protocoles IPv4 et IPv6.

1.4.2 Zigbee

C'est une solution intégrale qui opère généralement en conformité avec la norme IEEE 802.15.4 et exploite de réseau maillé sans fil. C'est une technologie sans fil fondée sur des standards élaborés pour favoriser les réseaux M2M et l'Internet des objets à faible coût et basse consommation d'énergie. ZigBee propose aux fabricants de matériel et d'applications IoT, ainsi qu'aux utilisateurs finaux, une grande flexibilité et une compatibilité remarquable. Le protocole a diverses applications grâce à ses caractéristiques distinctives, notamment un coût réduit, une faible consommation énergétique et une connectivité sans fil efficace. Par exemple, c'est une technologie privilégiée pour les maisons intelligentes puisqu'elle surpasse les autres technologies sans fil à divers aspects.

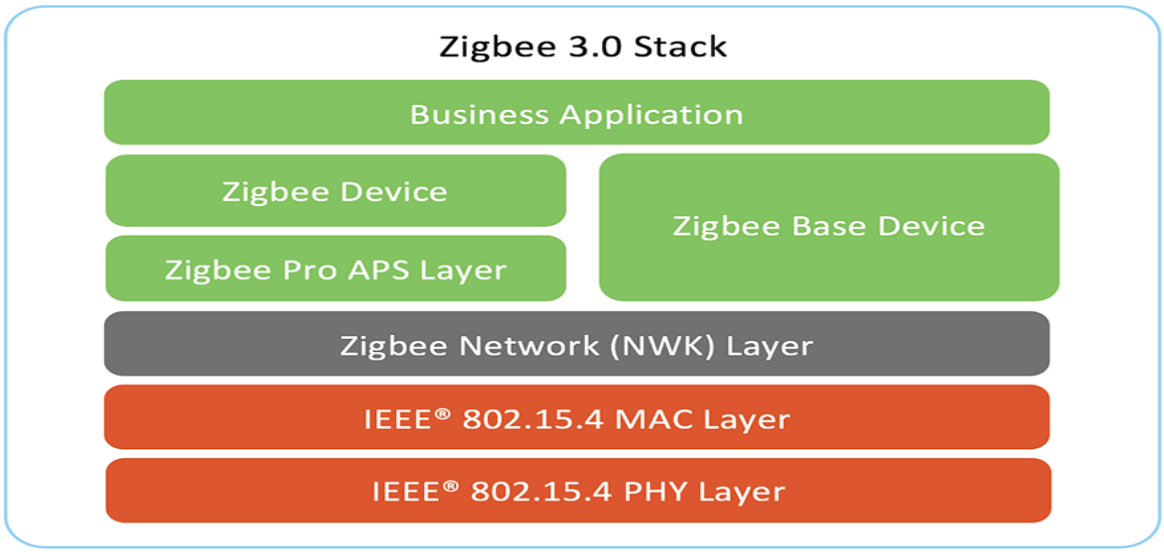

Figure 1.1 : Caractéristique de la technologie Zigbee.

1.4.3 Bluetooth Low Energy

Le Bluetooth Low Energy est une variante à faible consommation d'énergie de la technologie Bluetooth de réseau personnel (PAN), destinée à être utilisée par les machines et appareils connectés à Internet.

Également connu sous le nom de Bluetooth Smart, il a été intégré à la spécification Bluetooth 4.0 comme une alternative au Bluetooth Classique. Tout comme son prédécesseur, il exploite la technologie sans fil à saut de fréquence dans la bande radio non autorisée de 2,4 GHz pour établir une connexion entre des appareils à proximité. Contrairement à son prédécesseur, le Bluetooth LE atteint une vitesse maximale de seulement 1 Mbps tout en consommant entre 0.01 et 0.5 watts. Cela représente jusqu'à un tiers de la vitesse du Bluetooth Classique, tout en consommant pas plus de la moitié de l'énergie.

1.5 TECHNOLOGIES DE COURTE PORTEE

1.5.1 NFC

NFC (Near Field Communication) signifie Communication en Champ Proche. Les détails de la spécification NFC sont disponibles dans l'ISO 18092. La principale caractéristique de la NFC est qu'il s'agit d'une interface de communication sans fil avec une portée limitée à environ 10 cm. L'interface peut fonctionner en plusieurs modes. On fait la distinction entre les modes en fonction de si un appareil génère son propre champ RF ou s'il tire l'énergie du champ RF créé par un autre appareil. Si l'appareil produit son propre champ, il est qualifié de dispositif actif ; sinon il est désigné comme un dispositif passif. Les dispositifs actifs nécessitent généralement une alimentation électrique, tandis que les dispositifs passifs en ont rarement besoin (par exemple, la carte intelligente sans contact). Lors de la communication entre deux appareils, trois configurations distinctes sont envisageables. Tableau I.1 en présente la description [1].

Appareil A	Appareil B	Description
Active	Active	Lorsqu'un appareil transmet des données, il crée un champ RF. Lorsqu'il attend des données, un appareil ne produit pas de champ RF. Par conséquent, le champ RF est tour à tour produit par l'Appareil A et l'Appareil B.
Active	passive	Seul l'appareil A génér le champ RF
Passive	Active	Seul l'appareil B génér le champ RF

Tableau 1.1 : Configurations d'appareils actifs et passifs pour la génération de champ RF.

1.5.2 RFID

Les technologies d'identification automatique représentent une méthode novatrice de gestion des flux d'information et de matériaux, particulièrement adaptée aux vastes réseaux de production. La technologie RFID est un moyen de recueillir des informations sur un certain objet sans avoir à le toucher ou à voir le support de données, grâce à l'utilisation d'un couplage inductif ou d'ondes électromagnétiques. Le support de données est une micro-puce fixée à une antenne (ensemble appelé transpondeur ou étiquette) ;

Une caractéristique essentielle qui rend possible l'utilisation de la RFID pour le suivi des objets est sa capacité à offrir une identification unique. Une méthode possible pour identifier les articles est le Code Produit Électronique (EPC). Ce dernier offre un numéro standardisé au sein du réseau EPC global, avec un Service de Nom d'Objet

(ONS) fournissant les adresses web appropriées pour consulter ou mettre à jour des données spécifiques à une instance . Pour l'instant, ONS ne peut pas être utilisé dans un environnement mondial et étant donné que c'est un service propriétaire, son coût d'utilisation peut être assez élevé, en particulier pour les participants disposant de ressources limitées comme les PME (Petite et Moyenne Entreprise).

La technologie RFID repose sur deux principaux composants : le transpondeur (ou étiquette), qui contient une micro-puce pour stocker les données et une antenne pour les transmettre, et le lecteur RFID, qui envoie des ondes radio pour activer l'étiquette et lire les informations en quelques millisecondes. Grâce à sa rapidité et sa fiabilité, la RFID est largement utilisée dans des domaines variés tels que la logistique (pour le suivi des stocks), le contrôle d'accès (authentification des utilisateurs) et le commerce de détail (optimisation de la gestion et de l'expérience client).[2]

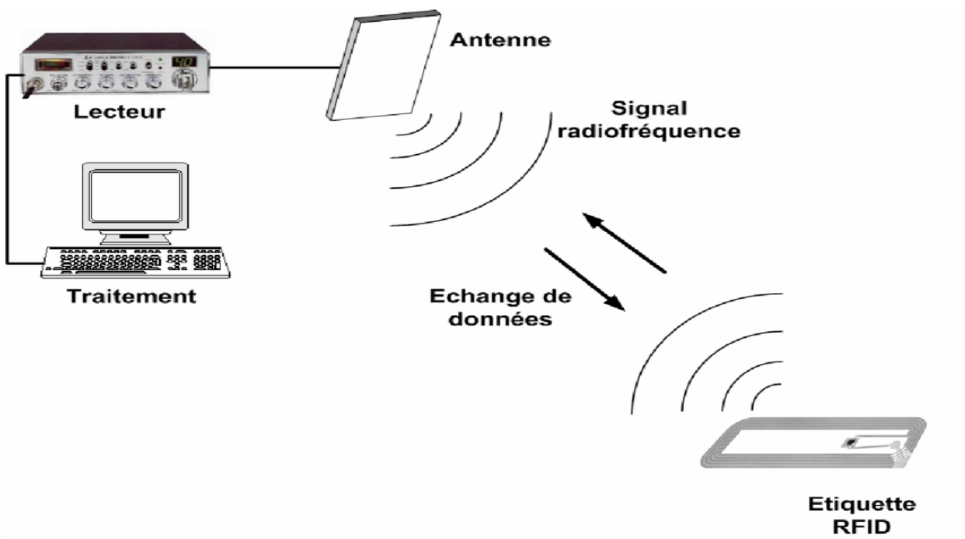

Figure 1.2 : Composants de la RFID.

1.6 ELEMENTS ESSENTIELS D'UN SYSTEME IOT

- **Acquisition de données** : Les objets connectés sont dotés de capteurs qui leur permettent de recueillir en permanence un flux constant de données. Ces données peuvent inclure des informations telles que la température ambiante, la pression atmosphérique, la position géographique, les mouvements détectés, ainsi que l'état de fonctionnement de l'appareil. Les capteurs intégrés aux objets connectés assurent une collecte continue et diversifiée de données, offrant ainsi une représentation précise de l'environnement et de l'activité de l'objet.

- **Gestion et étude des données** : Les informations recueillies par les appareils connectés sont traitées et examinées soit en temps réel, soit par agrégation, afin d'en dégager des renseignements pertinents. Ce processus peut inclure le traitement des données, leur consolidation, l'analyse statistique ainsi que l'application de techniques d'apprentissage automatique.

- **Représentation et intervention :** Les données extraites sont présentées sous divers formats tels que des tableaux de bord, des graphiques, des alertes ou d'autres représentations, afin de faciliter leur compréhension par les utilisateurs et de favoriser une prise de décision éclairée. Il est également possible de déclencher des actions automatisées basées sur l'analyse des données.

- **Administration et entretien :** Les dispositifs IOT requièrent une surveillance et un entretien réguliers pour assurer leur performance optimale et la protection des données. Cela peut inclure des mises à jour logicielles, des correctifs de sécurité, la supervision de l'état des équipements, ainsi que le remplacement

des accumulateurs. L'IOT fonctionne selon un cycle continu de recueil de données, de communication, de traitement, de représentation et d'action, favorisant ainsi une interaction intelligente entre le monde réel et l'univers digital.

1.7 TYPES DES OBJETS CONNECTES

1.7.1 Objet connecté simple

Détecteurs de mouvement : Ces dispositifs sont employés pour repérer le mouvement dans les résidences, les édifices, les espaces extérieurs et divers autres milieux. Ils peuvent servir à activer des systèmes d'alarme, à allumer les lumières de façon automatique, ou encore à suivre les déplacements d'individus ou d'animaux [3].

Détecteurs de pression : Ces dispositifs servent à mesurer la pression dans les pneumatiques, les conduites, les systèmes hydrauliques et d'autres usages. On peut les employer pour surveiller le rendement des machines, détecter les fuites ou les dysfonctionnements, ou contrôler les procédures industrielles.

Figure 1.3 : Capteur de mouvement

Détecteurs de température : Ces dispositifs servent à mesurer la température au sein des habitations, des édifices, des installations industrielles et d'autres

environnements. Ils peuvent être utilisés pour réguler les systèmes de chauffage et de refroidissement, surveiller l'état de conservation des aliments ou des produits pharmaceutiques, ou encore identifier les problèmes en matière de sécurité.

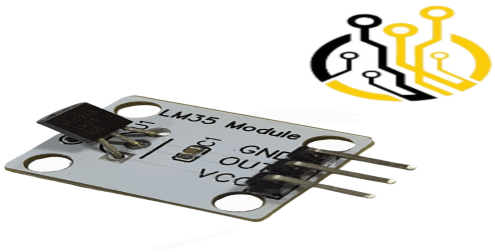

Figure 1.4 : Capteur de LM35

1.7.2 Objet connecté intelligent

Serrures intelligentes : Grâce aux serrures intelligentes, il est possible de verrouiller et déverrouiller une porte à distance via un smartphone ou une tablette [3]. Il est également possible de les configurer pour qu'elles s'ouvrent automatiquement dès que l'utilisateur s'approche de la porte.

Figure 1.5 : Serrure intelligents

Les serrures intelligentes renforcent la sécurité grâce à des fonctions avancées telles que le chiffrement des données, la détection d'intrusions et les alertes en cas de comportements suspects. Compatibles avec divers systèmes de domotique, elles permettent une gestion de l'accès via des appareils connectés comme les assistants vocaux ou les alarmes. Offrant un accès sans clé grâce à des codes, des empreintes digitales ou des applications mobiles, elles permettent également de verrouiller ou déverrouiller la porte à distance, assurant un contrôle total même en cas d'absence.

- **Appareils électroménagers :** Les équipements ménagers intelligents, comme les réfrigérateurs, les machines à laver et les sèche-linges, sont contrôlables à distance via un téléphone intelligent ou une tablette. Ils peuvent aussi être configurés pour commencer ou arrêter automatiquement à des heures déterminées.

- **Autres appareils**

    - **Aspirateurs :** Les robots aspirateurs ont la capacité de nettoyer les surfaces de façon autonome, en évitant les obstacles et en regagnant leur station d'accueil pour se recharger.

    - **Appareils à café :** Ils peuvent être programmés pour préparer le café à une heure spécifique, et certains d'entre eux sont même contrôlables par commande vocale.

    - **Thermostats :** Ces dispositifs peuvent être commandés à distance pour moduler la température de l'habitat, et certains types sont capables

d'apprendre les préférences des résidents et de s'ajuster de manière autonome. Ils peuvent également apprendre les habitudes des utilisateurs et ajuster automatiquement la température en conséquence.

Figure 1.6 : Appareils électroménagers intelligents

1.8 ARCHITECTURE D'UN SYSTEME IOT

Il n'y a pas d'architecture IoT unique établie et universellement reconnue, mais le format le plus élémentaire et largement accepté est une structure IoT à cinq couches qui inclut :

1.8.1 Couche de perception : La couche de perception de l'IoT transforme les signaux analogiques en informations numériques et réciproquement[5]. Elle utilise des dispositifs physiques intelligents tels que des capteurs et des étiquettes RFID pour recueillir des données du monde réel et permettre la communication entre les dispositifs, ce qui contribue à l'optimisation des opérations.

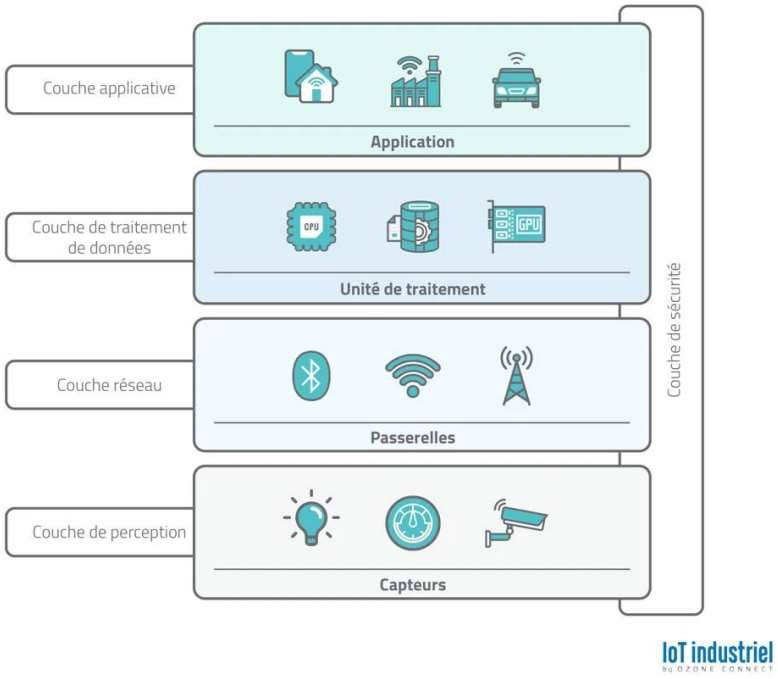

Figure 1.7: Architecture d'un système IoT.

1.8.2 Couche réseau : Cette couche assure la liaison entre les appareils et les objets connectés, les serveurs ainsi que d'autres équipements de réseau afin de gérer efficacement les données recueillies. Elle est responsable de la gestion du transfert de données, y compris les passerelles Internet et réseau, qui garantissent l'interconnexion entre les réseaux de capteurs et Internet. Les systèmes d'acquisition de données, également connus sous le nom anglais de Data Acquisition Systems (DAS), consolident et transforment aussi les informations à ce niveau.

1.8.3 Couche de traitement des données : La couche de traitement remplit les rôles principaux :

- Gérer les données en temps réel issues de la couche réseau pour permettre à l'administrateur de juger leur pertinence et emplacement.
- Stocker les informations pertinentes et véritablement utiles dans différentes options de stockage [4].

1.8.4 Couche Applicative : Elle établit la communication avec l'utilisateur via des services spécifiques. Les plateformes IoT peuvent fonctionner comme une infrastructure de développement logiciel, offrant des outils intégrés pour l'exploration des données, l'analyse détaillée et la visualisation des données. Ainsi, il est possible de concevoir des applications directement sur ces plateformes.

1.8.5 Couche de protection : Cette couche s'applique transversalement à toutes les couches antérieures. La sécurité dans l'IoT est d'une importance capitale [4].

1.9 DEFIS TECHNIQUES ET AVANTAGES

- **Évolutivité et Connectivité :** Alors que le nombre d'appareils IoT continue d'augmenter il est essentiel que le réseau soit capable de s'adapter pour gérer l'augmentation du trafic sans dégrader la performance [5].

- **Gestion des Données :** L'analyse en temps réel du Big Data produit par l'IoT exige des plateformes performantes et sophistiquées pour gérer ces grandes quantités de données.

- **Respect des Standards Techniques :** Les créateurs sont tenus de répondre à plusieurs normes techniques tout en réduisant la consommation d'énergie et en minimisant les interférences électromagnétiques entre les dispositifs.

- **Intelligence artificielle (IA) afin d'améliorer la sécurité :** Recourir à l'intelligence artificielle pour identifier rapidement toute irrégularité dans le système peut contribuer à améliorer sa protection globale contre les menaces et vulnérabilités du système IoT.

- **Sécurité :** Chaque dispositif connecté représente une possible vulnérabilité en matière de sécurité [6], exigeant une défense solide contre les cyberattaques et autres intrusions.

- **Efficacité et Automatisation :** L'IoT facilite l'automatisation de diverses tâches, augmentant ainsi la productivité tout en diminuant les dépenses opérationnelles.

- **Optimisation des Processus :** Dans le secteur industriel, l'IoT améliore la maintenance préventive en s'appuyant sur les informations recueillies par les capteurs, ce qui diminue de ce fait les arrêts imprévus.

- **Innovation à travers différents domaines :** L'IoT révolutionne divers domaines tels que l'agriculture (arrosage intelligent), les smart cities (contrôle de la circulation) et la maison intelligente.
- **Amélioration de l'expérience client :** En utilisant des analyses prédictives basées sur les données IoT, les entreprises ont la capacité de mieux anticiper les attentes de leurs clients.

1.10 CONCLUSION

Cette expansion rapide et l'adoption à grande échelle de l'Internet des objets (IoT) dans des secteurs aussi vitaux que la santé, l'automobile, l'agriculture et l'éducation, tout en ouvrant un champ immense d'applications commerciales, ont inévitablement soulevé une préoccupation majeure : la sécurité. La prolifération des dispositifs connectés, souvent avec des capacités de sécurité limitées, a créé un terrain fertile pour de multiples défis et vulnérabilités. Ces faiblesses potentielles peuvent être exploitées pour compromettre la confidentialité des données, perturber le fonctionnement des systèmes et même mettre en danger la sécurité physique des individus. Par conséquent, il est impératif de se concentrer sur le développement et l'implémentation de mesures de sécurité robustes et adaptées pour garantir un déploiement sûr et bénéfique de l'IoT à l'avenir.

CHAPITRE 2

ENJEUX DE SECURITE SPECIFIQUE A L'INTERNET DES OBJETS

2.1 INTRODUCTION

La mise en place de mesures critiques de sécurité des informations relatives à l'Internet des objets (IoT) est fondamentale pour diverses fonctionnalités, y compris, mais sans s'y limiter, l'identification et la confidentialité des données. Par conséquent, le développement rapide et l'intégration de multiples appareils aboutissent à la création d'une vaste infrastructure IoT. Il est donc prévu que l'écosystème IoT sera confronté à des menaces liées à sa technologie multiforme et à ses avancées potentielles. Les vulnérabilités de sécurité dans le contexte de l'IoT, telles que le déni de service, les attaques par force brute, les attaques de type man-in-the-middle, ainsi qu'une multitude d'autres menaces, font actuellement l'objet d'un examen minutieux dans le cadre du paradigme des réseaux interconnectés. La prévalence de ces attaques peut être attribuée à divers facteurs,tels que la sécurité insuffisante des mots de passe, des protocoles de cryptage inadéquats et la fuite involontaire d'informations personnelles, ce qui rend le stockage de ces données sensibles dans des environnements cloud particulièrement préoccupant. Si ces failles de sécurité ne sont pas corrigées dans un cadre de sécurité acceptable, les défaillances des services de sécurité qui en résulteront pourraient présenter des risques importants pour le paysage du marché de l'IoT. Cette situation englobe non seulement les défis de sécurité susmentionnés, mais également les problèmes liés au contrôle d'accès, aux processus d'authentification sur divers réseaux et aux complications associées au stockage des données.

2.2 VULNERABILITES ET MENACES LIEES A LA SECURITE DE IOT

2.2.1 Modèle de menaces

Une fois que vous avez obtenu une quantité suffisante d'informations pertinentes pour le test d'intrusion, l'étape suivante consiste à repérer les diverses menaces qui pourraient impacter le sujet. Le modèle agit en tant que prototype pour l'observateur. Il expose les faiblesses du système tout en établissant la méthode d'attaque. On évalue ensuite une multitude de scénarios pour traiter la question. Il est crucial d'être aussi complet que possible et d'identifier le schéma d'assaut le plus performant contre le système concerné. Pour chaque menace détectée, il est crucial de juger sa réalisabilité en détaillant les diverses méthodes utilisées pour la mettre en œuvre.

2.2.2 Analyse des vulnérabilités

Suite à l'identification des éléments du système et à la création du panorama des menaces, la phase tertiaire déclenche le processus de détection et d'étude des vulnérabilités éventuelles présentes dans le système. C'est à ce point que l'auditeur de sécurité examine la facilité d'application des divers scénarios élaborés. En s'appuyant sur les données recueillies et les faiblesses détectées, l'évaluateur juge la viabilité de l'attaque envisagée. Ainsi, à l'issue de cette analyse, seuls les scénarios d'agression qui présentent des possibilités de réussite significatives demeurent [7] .

2.3 EXPLOITATION

À ce stade, l'expert en tests d'intrusion exploite les différentes vulnérabilités pour valider leur existence authentique et garantir qu'il ne s'agit pas de faux positifs. Cette

étape est essentielle, tout en étant sensible et nécessitant une compétence approfondie ainsi qu'une compréhension complète des mesures appliquées, dans le but de réduire au minimum les potentiels impacts négatifs. En effet, si elle n'est pas correctement gérée, l'exploitation d'une vulnérabilité peut entraîner un fonctionnement défectueux, voire une panne totale du système. Lorsque l'exploitation d'une vulnérabilité réussit, le système se trouve alors compromis. L'intrus a désormais le moyen d'entrer dans le système ou d'interagir avec lui [6].

2.4 APRES L'EXPLOITATION

Selon les privilèges spécifiques obtenus, les buts à réaliser varient ; toutefois, la méthode pour parvenir à ces buts resterà constante. Effectivement, pour obtenir des privilèges supplémentaires ou accéder à des renseignements confidentiels, l'auditeur doit une fois de plus rassembler des données en prenant en compte les informations fraîchement obtenues.

Comme le démontre la figure 2.1, nous sommes à l'aube du début d'un nouveau cycle. L'auditeur effectue autant de cycles que nécessaire jusqu'à l'atteinte de l'objectif.

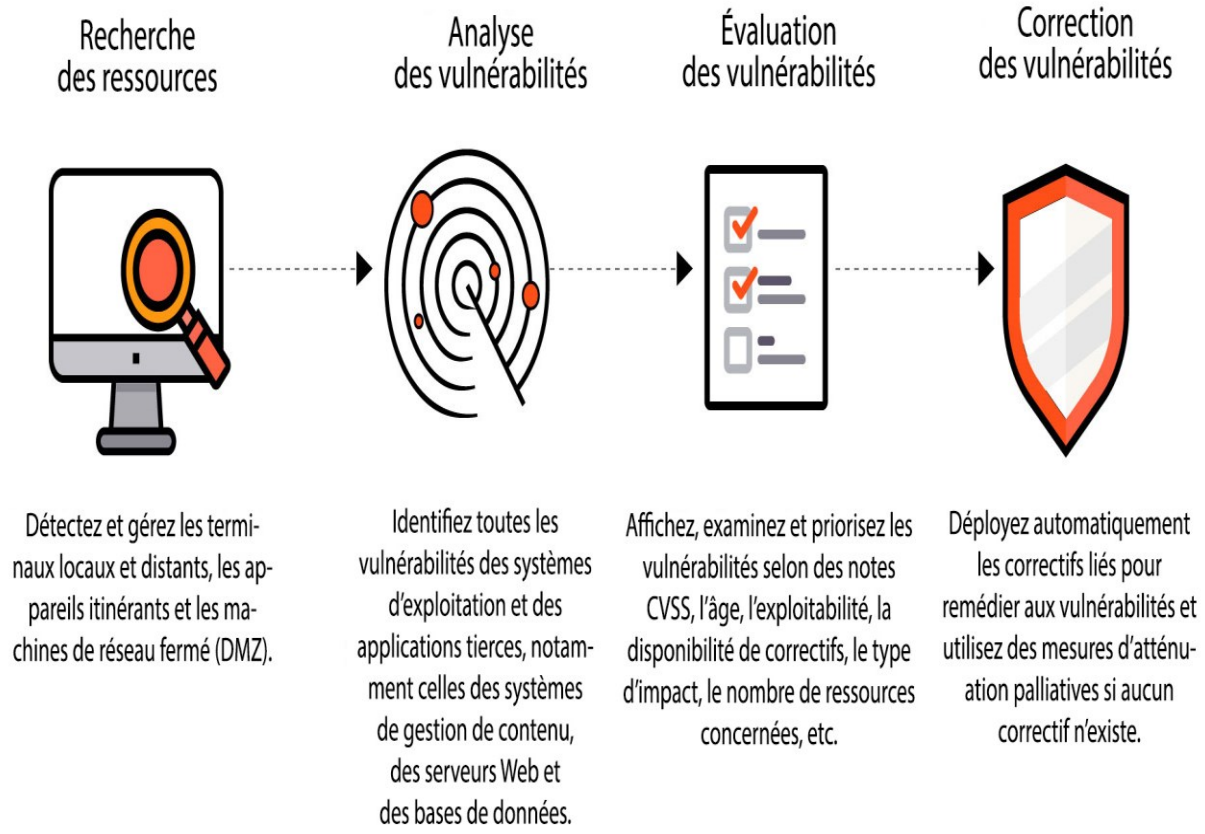

Figure 2.1 : Vulnérabilités et menaces liées à la sécurité

2.5 VULNERABILITES ET MENACES LIEES A LA SECURITE DE IOT

- **Attaque DDos**

Une attaque par déni de service distribué (DDoS) est une variété de cyberattaque qui implique l'utilisation de plusieurs ordinateurs infectés, chacun étant contaminé par un malware. On se sert de ces ordinateurs pour inonder un système ou un réseau particulier avec une masse considérable de trafic ou de demandes. L'objectif d'une attaque DDoS est de rendre le système ou le réseau visé inopérant, bloquant ainsi

l'accès pour les utilisateurs autorisés. Ces attaques peuvent se produire partout dans le monde et il est souvent difficile de les identifier et de les prévenir [6].

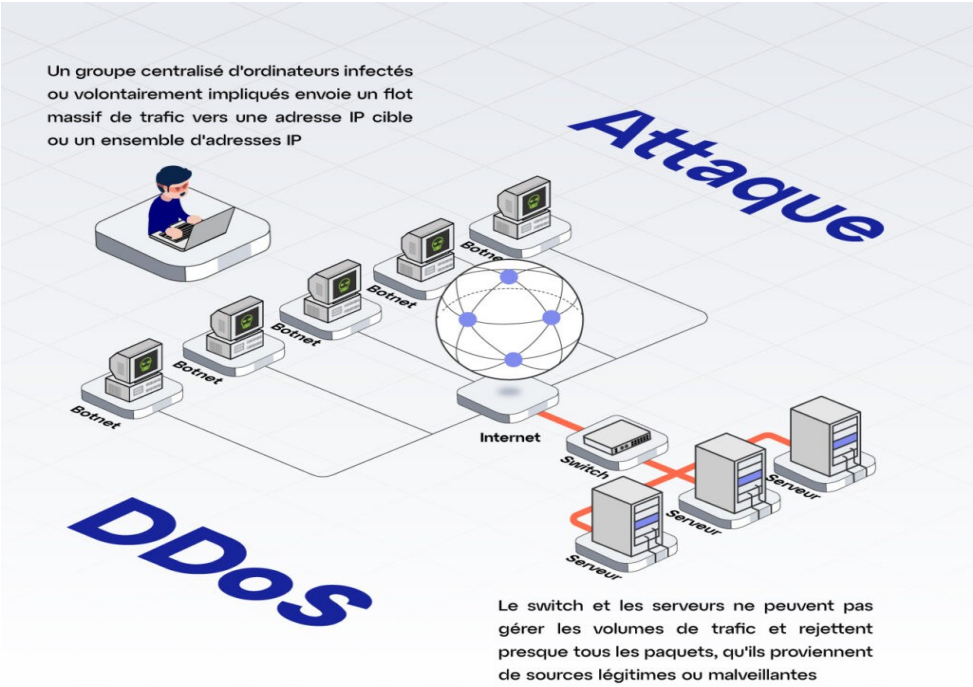

Figure 2.2 : Illustration d'une attaque DDos

Les attaques DDoS se manifestent sous plusieurs formes. Parmi les plus courantes, on distingue :

- **Attaques de la couche applicative :** Ces attaques ciblent les ressources ou les services d'une application, empêchant les utilisateurs légitimes d'y accéder. Elles sont généralement mesurées en nombre de requêtes par seconde. Les attaques typiques incluent le déni de service HTTP GET et HTTP POST.
- **Attaques de protocole :** Ces attaques exploitent **les tables d'état des équipements réseau** dédiés aux échanges de données, tels que les équilibreurs de charge, les pare-feu et les serveurs consacrés aux applications. **L'attaque**

est quantifiée en paquets par seconde : attaques TCP, SYN, ACK, ainsi que d'autres attaques survenant durant le protocole.

- **Attaques par saturation :** L'attaque par saturation cherche à saturer la capacité de bande passante du site ciblé, et s'évalue en bits par seconde. Cela comprend des techniques telles que l'inondation ICMP (Internet Control Message Protocol), l'inondation UDP, ainsi que d'autres méthodes similaires.

- **Perturbation du nœud dans le réseau de capteurs sans fil (Jamming) :** Dans les réseaux de capteurs sans fil (WSN), les attaques de brouillage sont effectuées par des nœuds malveillants, dans le but de perturber ou d'entraver l'envoi et la réception de signaux sans fil légitimes entre les capteurs. Les attaques de brouillage affectent les paramètres statistiques (tels que la moyenne et la variance) d'un flot de paquets en présence de variations temporelles. On peut donc recourir à la maîtrise statistique des procédés (SPC) pour identifier cette irrégularité en analysant des séquences d'événements statistiquement uniformes.

2.5.1 Attaques sur les protocoles IoT

- **MQTT Dos :** MQTT est un protocole de messagerie léger fréquemment utilisé dans le cadre de l'Internet des Objets (IoT). Bien qu'il soit efficace, il n'est pas à l'abri des menaces de sécurité, y compris les attaques par déni de service (DoS) [7]. Une attaque DoS contre un courtier MQTT vise à perturber son fonctionnement normal en l'inondant de trafic ou de requêtes malveillantes. Cela peut provoquer un manque de réactivité du courtier, empêchant les clients légitimes de se connecter ou d'échanger des messages.

- **Attaque par amplification CoAP :** Une attaque d'amplification CoAP représente une forme d'attaque par déni de service distribué (DDoS) exploitant le Protocole contraint d'application (CoAP) afin d'intensifier le trafic malveillant et de saturer une cible.

  — **Requêtes malveillantes :** L'attaquant envoie des requêtes CoAP à plusieurs serveurs CoAP ouverts, conçues pour provoquer des réponses volumineuses.

  — **Amplification :** Les serveurs CoAP répondent avec des réponses plus volumineuses que les requêtes initiales, créant un effet d'amplification.

  — **Inondation de la cible :** Les réponses amplifiées sont dirigées vers la cible, la submergeant avec un volume massif de trafic.

- **Attaque sur 6LoWPAN** : Il s'agit d'une technologie conçue pour faire fonctionner le protocole IPv6 sur des réseaux sans fil à faible consommation d'énergie, généralement utilisés dans le domaine de l'Internet des objets. Malgré ses nombreux atouts, 6LoWPAN présente une susceptibilité à diverses formes d'attaques.

2.5.2 Attaques via L'IoT Edge Computing

L'informatique en périphérie, également appelée Edge Computing, est une stratégie qui vise à traiter les informations à proximité de leur origine, soit directement sur les dispositifs IoT (Internet des objets), plutôt que de les transférer

vers un serveur cloud centralisé. Bien que cette méthode offre de multiples bénéfices concernant la latence, le débit et la confidentialité, elle soulève également de nouveaux enjeux en termes de sécurité [7].

- **Sécurité matérielle des équipements :** Assurer une protection physique des dispositifs IoT pour prévenir le vol, la manipulation ou la destruction.

- **Mises à jour fréquentes des logiciels et firmwares :** Assurer la mise à jour des logiciels et firmwares des dispositifs IoT avec les correctifs de sécurité les plus récents.

- **Communication en toute sécurité :** Il est essentiel d'établir des méthodes solides de cryptage et d'authentification pour sécuriser les échanges entre les dispositifs IoT.

- **Contrôle et suivi du trafic :** Examiner le trafic réseau afin d'identifier les irrégularités et les comportements suspects, tout en instaurant des procédures de filtrage pour empêcher les requêtes malveillantes.

- **Segmentation du réseau :** Pour réduire l'impact d'une éventuelle compromission, il est conseillé de séparer les divers types d'appareils IoT sur des sous-réseaux distincts.

- **Gestion des accès et des identités :** Utilise des systèmes d'authentification et d'autorisation solides pour réguler l'accès aux dispositifs IoT et aux serveurs edge.

- **Détection et intervention en cas d'incident :** Établir des systèmes pour détecter les intrusions et réagir aux incidents afin de repérer rapidement les attaques et d'intervenir efficacement.

2.6 SOLUTIONS DE SECURITE CLASSIQUES POUR L'IOT

2.6.1 Mécanismes de sécurité logiques

- **Confidentialité :** Le service de confidentialité offre une protection contre la divulgation des flux de données et l'analyse du trafic par des entités non autorisées. Le moyen de protection le plus approprié pour garantir ce service de sécurité est le cryptage des données. Cela peut être accompli avec un système soit asymétrique (utilisant une clé publique), soit symétrique (utilisant une clé secrète). Le système de cryptage symétrique repose sur la connaissance de la clé secrète utilisée pour le chiffrement et le déchiffrement. Néanmoins, en ce qui concerne le système de cryptographie asymétrique, le fait que toutes les parties aient accès à la clé publique de chiffrement ne signifie pas nécessairement qu'elles détiennent également la clé privée associée. Outre les mécanismes de cryptage, la présence d'un système de gestion des clés est indispensable.

- **Intégrité :** Dans le cadre de l'Internet des Objets, l'intégrité, qui est un service de sécurité, couvre deux aspects majeurs : l'intégrité des données et l'intégrité des objets. L'intégrité des données garantit que les informations transmises

dans un environnement IoT n'ont pas été modifiées ou supprimées de manière non autorisée pendant leur transit. Ceci est indispensable pour garantir la légitimité des commandes reçues par les objets et des informations collectées, ce qui permet d'offrir un service de confiance. La vérification de l'intégrité des données implique deux procédures distinctes : une du côté de l'émetteur et l'autre du côté du récepteur. Des données de vérification (comme un hachage ou un code de contrôle par bloc tel que le BCC, qui est une valeur de contrôle cryptographique) sont utilisées pour valider l'intégrité.

- **Disponibilité :** fait référence à la capacité d'une entité autorisée, sur requête, d'avoir accès à des ressources et de les exploiter suite à une authentification et un contrôle d'accès. Un service qui, par exemple, devient indisponible suite à une attaque de type DoS est un service susceptible d'être compromis à tout moment, donc non protégé. Ainsi, la disponibilité est considérée comme un service de sécurité essentiel. Pour assurer un environnement parfaitement fonctionnel en plus de la connectivité Internet, la présence continue sur l'Internet des Objets est primordiale. Ce service, d'une part, garantit l'accessibilité permanente du service IoT pour les utilisateurs, et d'autre part, il permet une collecte de données ininterrompue en tenant compte de la disponibilité des appareils (c'est-à-dire, objets, passerelles).

- **Identification et Authentification :** Cette démarche repose sur l'usage d'un identifiant spécifiquement attribué à chaque utilisateur, visant à établir l'identité de l'utilisateur d'un service IoT. L'authentification, qui permet à l'utilisateur de prouver son identité, intervient après l'étape d'identification. L'utilisateur utilise un code secret ou un identifiant auquel seul lui a accès. En réalité, c'est le contrôle d'accès qui accorde un droit d'entrée si

l'authentification a été réussie ; donc ce n'est pas cette dernière qui garantit ce privilège. Les mécanismes d'identification et d'authentification renforcent la sécurité de l'environnement IoT. Ainsi, grâce à ces deux éléments, cet environnement incorpore des mécanismes solides et aptes à prévenir les infractions et à minimiser les risques d'intrusion.

- **Gestion des accès :** Le contrôle d'accès empêche une utilisation non autorisée d'une ressource IoT. Pour mettre en œuvre ce contrôle, une liste des entités autorisées à accéder à une ressource et leur niveau d'accès sont spécifiés conformément à une politique de sécurité. Cette prestation a pour objectif de permettre divers niveaux d'accès aux ressources (écriture, lecture, modification, exécution d'une tâche ou suppression d'une information). L'accès est régulé par le biais de bases d'informations détenues par la ressource ou par des centres d'autorisation, à partir d'un ou plusieurs critères (une matrice structurée distribuée ou hiérarchique, une liste de contrôle d'accès, etc.). Ces bases d'informations contiennent des données d'authentification, telles que des identifiants, des mots de passe, ou des étiquettes de sécurité, etc.

Services de sécurité	Mécanismes de sécurité	Quelques exemples
Confidentialité	Chiffrement des messages / cryptage des signes	Mécanismes cryptographiques symétriques (AES, CBC, etc), mécanismes cryptographiques asymétriques (RSA, DSA, IBE, ABE, etc).
Intégrité	Fonctions de hachage, signature des messages	Fonctions de hachage (SHA-256, MD5, etc), Codes d'authentification des messages (HMAC)
Authentification	Chaîne de hachage, Message code d'identification	HMAC,CBC-MAC, ECDSA
Non-répudiation	Signature des messages	ECDSA, HMAC
Disponibilité	Pseudo-aléatoire saut de fréquence, Contrôle d'accès, Prévention des intrusions systèmes, pare-feu	Signature basée sue les systèmes de détection d'intrusion Détection d'intrusion basée sur des anomalies statistiques

Tableau 2.1 : Services et mécanismes de sécurité logiques

2.6.2 Mécanismes de sécurité Réseaux

- **Software Defined Networking :** le SDN, qui est une nouvelle manière de construire, de concevoir, de sécuriser et d'exploiter les réseaux, est un nouveau concept émergent de gestion de ces derniers. Il fonctionne par le découplage du plan de contrôle, s'occupant uniquement de l'acheminement des paquets rendant les réseaux programmables et flexibles, et qui est responsable de l'association d'une décision de routage aux paquets du plan de données représentant l'infrastructure virtuelle ou physique, et ce, en étant basé sur une

gestion centralisée des flux réseaux. Avec le SDN, un équipement externe appelé « contrôleur », gérant les fonctions du plan de contrôle, gère et externalise l'intelligence du réseau.

- **Système de Détection/Prévention d'Intrusion (IDS/IPS) :** les systèmes de prévention ou de détection d'intrusion, ou encore IDS/IPS (Intrusion Detection/Prevention Systems) en anglais, sont des composants essentiels dans le cadre de la sécurité réseau. Ils ont pour objectif, par l'analyse des échanges au sein ou entre des systèmes, la prévention ou la détection d'intrusions. Pour ce, il faut que ces systèmes soient capables de capturer ces échanges, d'y identifier une attaque et d'empêcher la tentative d'intrusion et/ou de générer une alerte. Deux principaux types d'IDS/IPS sont identifiables :

  — IDS/IPS réseaux : Aussi appelés NIDS/NIPS (Network Intrusion Detection/Prevention Systems), ils fonctionnent à l'aide d'une sonde spécifique positionnée sur le réseau, et surveillent les échanges effectués sur celui-ci.

  — IDS/IPS hôtes : Aussi appelés HIDS/HIPS (Host Intrusion Detection/Prevention Systems), ils fonctionnent grâce à une ou plusieurs sondes implantées dans les hôtes, et surveillent la sécurité au niveau de ces derniers.

- **Cryptage basé sur l'identité (IBE)** : L'IBE représente un élément crucial de la cryptographie fondée sur l'identité. C'est une forme de cryptage à clé publique, où la clé publique d'un utilisateur sert d'identifiant unique lié à celui-ci (tel qu'une adresse email). L'algorithme d'IBE facilite une vérification rapide de la correspondance entre les paramètres publics et une clé privée, renforçant la confiance des utilisateurs envers les administrateurs du système. En outre, l'IBE offre la possibilité d'authentifier l'expéditeur du message à un coût de communication minime.

- **Cryptage basé sur les attributs (ABE)** : ont introduit le concept de chiffrement basé sur les attributs (ABE), comme une extension du chiffrement d'identité basé sur l'indétermination [8]. ABE met en place une méthode expressive de gestion de l'accès aux données confidentielles, en se servant d'une structure de politique d'accès qui détermine les relations entre un groupe d'attributs utilisés pour le chiffrement des données. Dans un système ABE, le serveur de production de clés produit pour chaque utilisateur autorisé une clé privée basée sur ses attributs, ainsi qu'une clé publique qui sert à chiffrer les données selon une politique déterminée [8] .

2.7 PROTECTION DE LA VIE PRIVEE

C'est l'une des méthodes les plus répandues dont la fonction principale est de préserver la confidentialité des flux de données. Le concept de cette approche est d'attribuer des labels additionnels, nommés tags, aux courants de données, afin de donner la possibilité aux systèmes informatiques fiables de traiter les flux de données confidentielles et ainsi dissimuler l'identité des personnes détenant ou supervisant

ces données. Toutefois, les systèmes de marquage peuvent représenter un défi pour les appareils limités, car la dimension des étiquettes s'accroît avec l'extension des données et engendre des calculs supplémentaires "coûteux".

- **Techniques de suivi IP**

Pour repérer les attaques par déni de service et les inondations IP en temps réel, les systèmes basés sur les techniques de suivi IP s'avèrent être puissants et sont couramment employés dans des réseaux IP comme Internet. Ces mécanismes ont pour mission primordiale d'augmenter la sécurité des protocoles légers, fondés sur l'IP et spécifiquement élaborés comme des versions adaptées des protocoles traditionnels TCP/IP pour l'Internet des Objets. Parmi les nombreux protocoles largement utilisés dans le domaine de l'IoT, RPL, 6LoWPAN et DTLS sont des exemples qui garantissent la disponibilité et l'intégrité des données transmises de manière intégrale entre les appareils IoT. Toutefois, ces protocoles ne sont pas à la base prévue pour traiter les attaques basées sur le DoS/DDoS.
La littérature propose diverses méthodes pour optimiser la couche de transport qui repose sur le protocole DTLS et le protocole de routage 6LoWPAN basé sur l'RPL. Ces dispositifs renforcent la solidité et l'efficacité des protocoles face aux actes malintentionnés. Ces solutions utilisent des passerelles IoT et des routeurs IP pour inspecter et étudier les paquets, afin de détecter les comportements nuisibles finaux et d'agir en conséquence.

2.8 ANALYSE DES APPROCHES DES IDS

Dans cette section, nous proposons une analyse structurée et méthodique des recherches actuelles. L'étude de la littérature nous a permis d'identifier plusieurs critères essentiels, évalués à l'aide de métriques clés, pour assurer une détection efficace des intrusions dans les environnements IoT.

- **Placement :** Tout comme pour les systèmes informatiques actuels, l'emplacement d'un système de détection d'intrusion est essentiel car il influence la mesure dans laquelle il peut exposer les activités du système surveillé. Par exemple, un IDS réseau ne peut que surveiller le trafic réseau généré ou destiné à l'hôte surveillé et, de ce fait, est incapable de détecter des activités malveillantes internes telles que des processus de subversion ou d'escalade de privilèges. Cela devient de plus en plus crucial pour les systèmes IoT, et des recherches récentes soulignent l'insuffisance de la sécurité des dispositifs IoT [9].

- **Délai de détection :** L'un des caractères distinctifs essentiels des systèmes IoT est leur nature dynamique, dans laquelle les nœuds participants opèrent selon un modèle ad hoc. Ainsi, la rapidité de détection prend une importance accrue lorsqu'il s'agit de détecter une attaque au plus vite pour empêcher sa propagation vers un plus grand nombre d'appareils. L'etude de la littérature existante nous a permis de constater que la plupart des méthodes actuelles sont mises en place et analysées dans des contextes isolés, ce qui engendre deux conséquences : premièrement, l'implémentation et l'évaluation ne reflètent pas fidèlement un environnement IoT ; deuxièmement, les résultats

obtenus à partir d'expérimentations effectuées dans des environnements simulés comme Matlab ou Contiki révèlent que ces simulations ne reproduisent pas précisément les défis auxquels on est confronté.

- **Moteur de détection :** Un IDS peut utiliser une variété de moteurs de détection, tels que les anomalies, les signatures ou les approches comportementales. Le choix du moteur de détection a un double impact, à savoir : il peut affecter la capacité d'un IDS à détecter des attaques, et il a un impact sur la surcharge de performances encourue par le moteur. Par exemple, bien que les IDS basés sur les signatures aient été identifiés comme étant économes en ressources, ils n'ont pas la capacité de détecter les attaques de type zero-day. Cependant, on observe une adoption croissante de techniques d'intelligence artificielle et d'apprentissage profond, permettant d'améliorer significativement l'efficacité de la détection des intrusions. [9]

2.9 CONCLUSION

Un IDS a la compétence d'utiliser différents moteurs de détection, tels que ceux basés sur les anomalies, les signatures et les collections de règles. Le choix du moteur de détection a deux conséquences cruciales : il peut affecter la compétence d'un IDS à détecter des attaques, et il a aussi une répercussion sur la surcharge de performances causée par le moteur. Par exemple, bien que les IDS basés sur des signatures soient jugés peu exigeants en termes de ressources, ils ne parviennent pas à détecter les attaques zero-day. Cependant, on observe également une augmentation de l'utilisation de l'intelligence artificielle et de l'apprentissage profond pour mettre en place une détection des intrusions performante.

CHAPITRE 3

SECURISATION DES RESEAUX IOT PAR DES TECHNIQUES D'INTELLIGENCE ARTIFICIELLE

3.1 INTRODUCTION

L'Internet des objets (IoT) joue un rôle important dans la transformation numérique dans de nombreux domaines. En interconnectant des capteurs, des instruments et divers appareils, l'IoT améliore l'acquisition et l'examen des données ainsi que la gestion automatisée. Le renforcement de la sécurité des réseaux IoT apparaît rapidement comme l'un des principaux défis auxquels est confronté le secteur des technologies de l'information. Néanmoins, compte tenu de l'évolution et du déploiement considérables des appareils IoT, la capacité de ces appareils à communiquer en toute sécurité sans compromettre les performances constitue un obstacle considérable. Par conséquent, l'adoption d'une stratégie globale de détection des intrusions est en train de devenir la principale solution de sécurité dans divers secteurs (transport, énergie, fabrication intelligente) afin de protéger les réseaux contre un large éventail d'activités malveillantes.

L'objectif principal de ce travail est d'utiliser un ensemble d'algorithmes d'apprentissage automatique et d'apprentissage profond pour identifier et prévoir efficacement les anomalies associées à diverses formes d'intrusions dans les réseaux de l'Internet des objets (IoT).

3.2 ALGORITHMES DE MACHINE LEARNING

Ils peuvent servir à repérer des a dans les données en identifiant le schéma sous-jacent et en décelant toute déviation de ce schéma. La section suivante est consacrée à la présentation des algorithmes d'apprentissage automatique les plus fréquemment utilisés pour détecter les attaques.

3.3 ARBRES DE DECISION

- **Les arbres de décision :** Les arbres de décision sont des modèles d'apprentissage supervisé qui prennent des décisions sous forme d'un arbre. Chaque nœud représente un test sur une caractéristique, chaque branche un résultat, et chaque feuille une prédiction. L'algorithme sélectionne à chaque étape la meilleure caractéristique pour diviser les données, en utilisant des mesures comme le gain d'information ou l'indice de Gini. Ce processus se répète jusqu'à ce qu'un critère d'arrêt soit atteint. Bien qu'ils soient faciles à interpréter, les arbres de décision peuvent surapprendre si l'arbre est trop profond. Ils sont souvent utilisés avec d'autres modèles, comme les forêts aléatoires, pour améliorer leur performance.

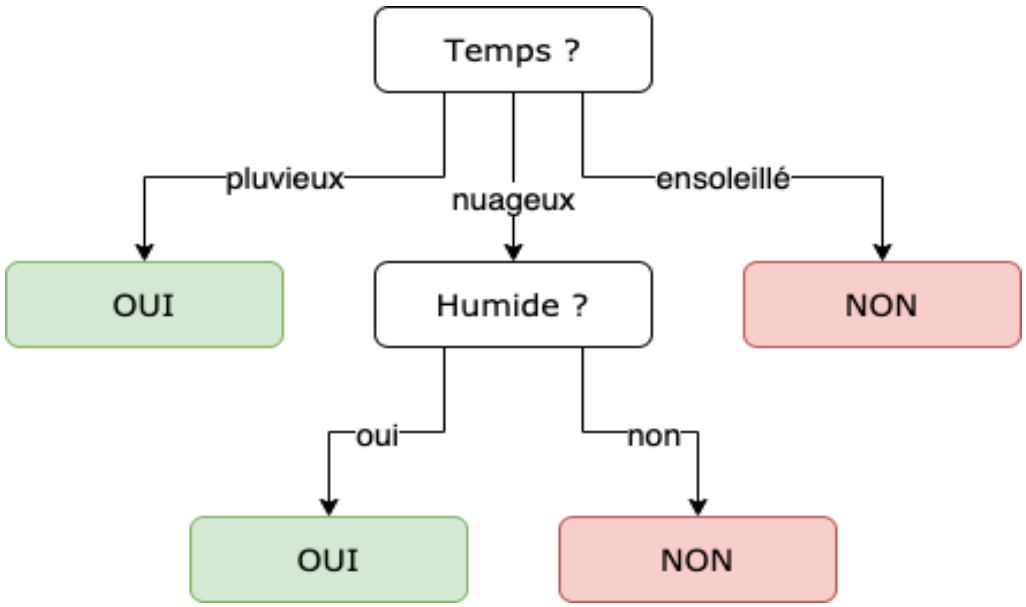

Figure 3.1 : Arbre de Décision

- **La machine à vecteurs de support (SVM) à classe unique :** est un modèle d'apprentissage supervisé utilisé pour la classification et parfois la régression. Son objectif est de trouver une ligne (ou un plan dans les cas plus complexes) qui sépare au mieux les différentes classes de données. Pour cela, il cherche

l'hyperplan qui maximise la marge, c'est-à-dire la distance entre ce plan et les points les plus proches de chaque classe (appelés vecteurs de support). Plus cette marge est grande, meilleure est la séparation. Quand les données ne sont pas séparables directement, SVM utilise une technique appelée le noyau (kernel) pour les transformer dans un espace de dimension supérieure où la séparation devient possible.

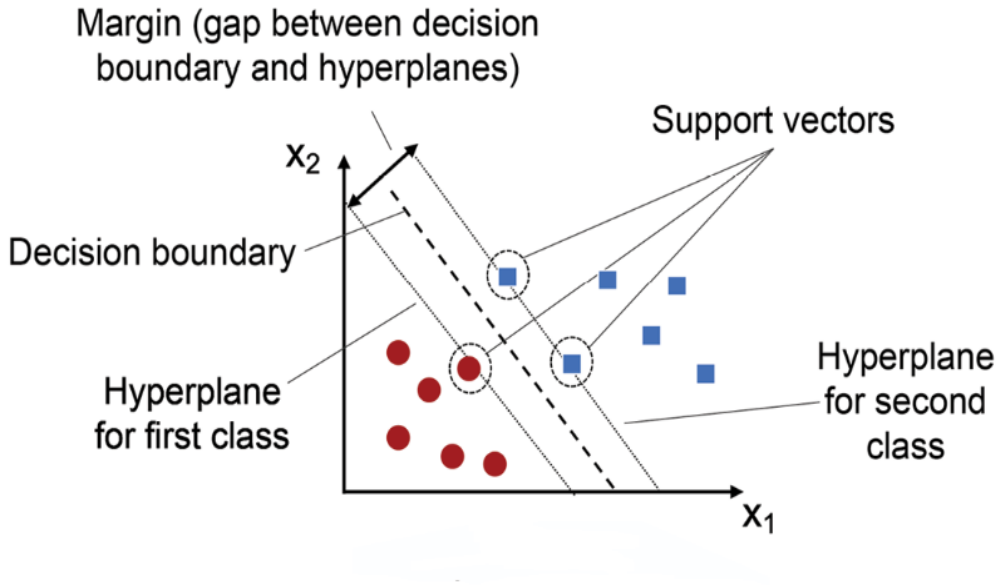

Figure 3.2 : Fonctionnement d'un Classifieur SVM

- **Les k plus proches voisins (k-NN) :** est une méthode d'apprentissage supervisé simple et intuitive utilisée pour la classification et la régression. Lorsqu'un nouvel exemple doit être classé, l'algorithme calcule la distance entre cet exemple et tous les exemples du jeu de données d'entraînement (généralement en utilisant la distance euclidienne). Ensuite, il sélectionne les k exemples les plus proches (voisins) de ce point. Pour la classification, il attribue à l'exemple la classe majoritaire parmi ses k voisins. Pour la régression, il calcule la moyenne des valeurs de ces voisins. Le choix du bon nombre de voisins (k) est important, car une valeur trop petite peut rendre le

modèle sensible au bruit, tandis qu'une valeur trop grande peut le rendre trop général. L'algorithme k-NN est facile à comprendre et à implémenter, mais il peut devenir lent avec de grandes bases de données, car il doit calculer la distance avec tous les exemples à chaque prédiction.

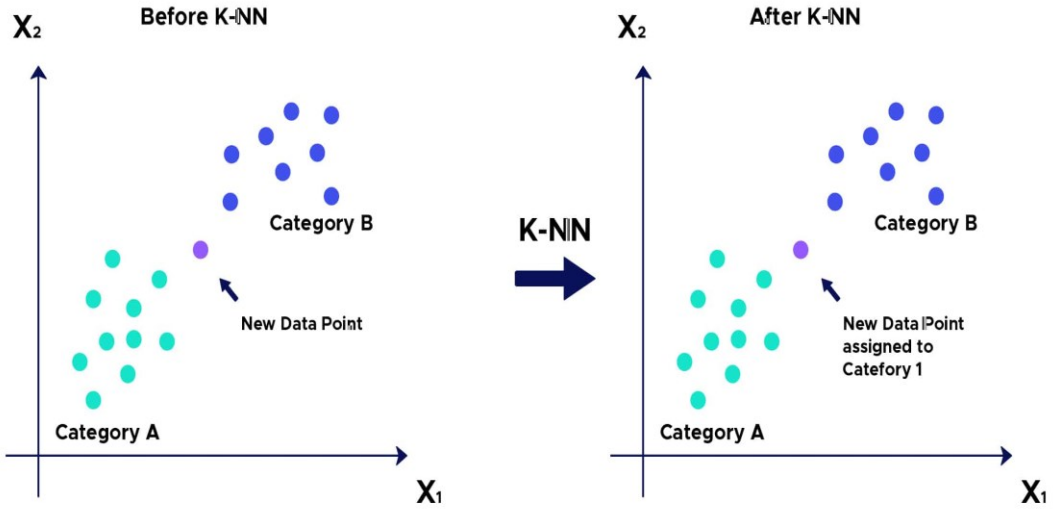

Figure 3.3 : Fonctionnement de l'Algorithme K-NN

- **Naive Bayesian :** est un modèle de classification basé sur le théorème de Bayes, qui calcule la probabilité qu'un exemple appartienne à une classe donnée. Il est dit "naïf" car il suppose que toutes les caractéristiques (attributs) sont indépendantes les unes des autres, ce qui est rarement vrai en pratique, mais fonctionne souvent très bien. Lorsqu'un nouvel exemple doit être classé, l'algorithme calcule la probabilité de chaque classe en fonction des valeurs de ses attributs, puis il choisit la classe ayant la probabilité la plus élevée. C'est un algorithme très rapide, efficace même avec peu de données, et particulièrement performant pour les textes (comme la détection de spam). Malgré sa simplicité, il donne souvent de bons résultats, surtout quand les variables sont nombreuses.

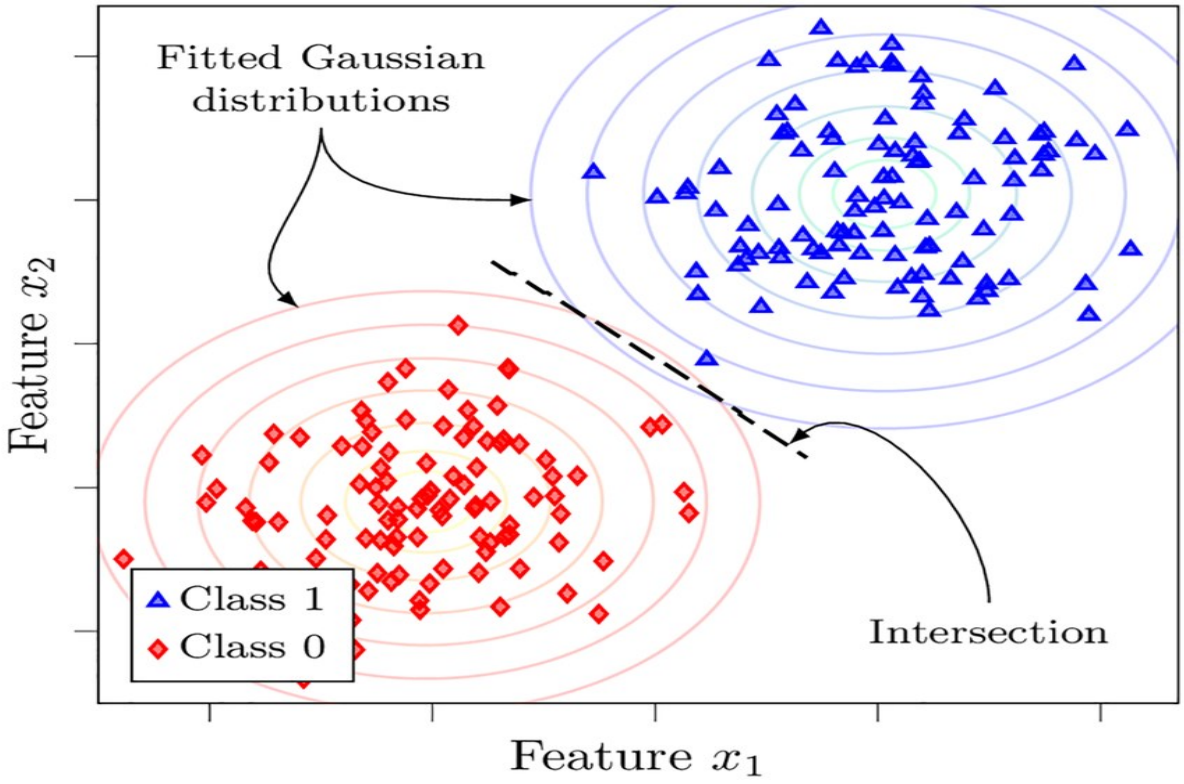

Figure 3.4 : Illustration de la classification bayésienne naïve

- **Logistic Regression:** est un algorithme d'apprentissage supervisé utilisé principalement pour la classification binaire, c'est-à-dire pour prédire si un exemple appartient à l'une ou l'autre de deux classes (par exemple : oui/non, 0/1). Contrairement à la régression linéaire, elle ne prédit pas directement une valeur, mais une probabilité. Elle utilise une fonction appelée sigmoïde pour transformer une combinaison linéaire des variables d'entrée (comme w1x1+w2x2+...+b) en une valeur comprise entre 0 et 1. Si cette probabilité est supérieure à un certain seuil (souvent 0.5), l'algorithme classe l'exemple dans la classe 1, sinon dans la classe 0. La régression logistique est simple, rapide à entraîner, et efficace pour des problèmes où les classes sont bien séparables. [10].

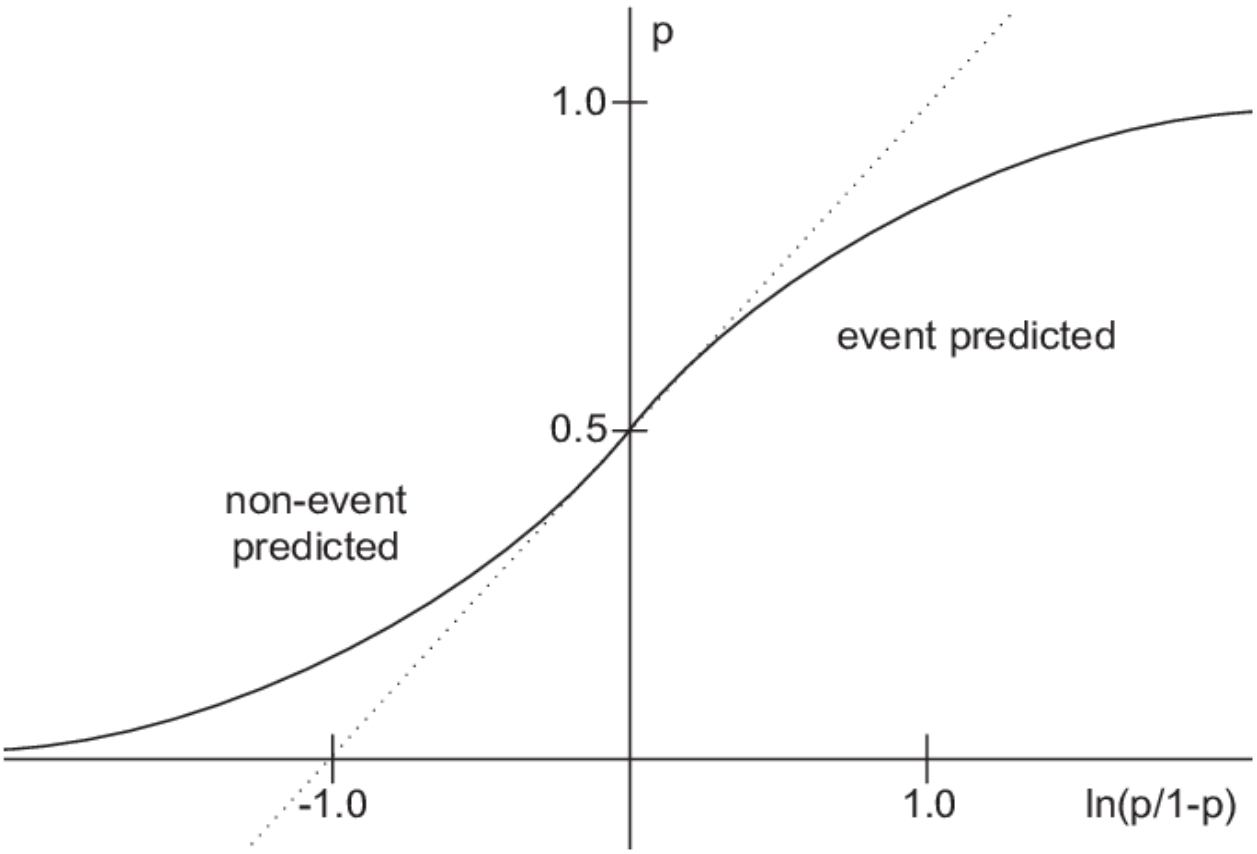

Figure 3.5 : Courbe de la régression logistique

- **Le facteur local de données aberrantes :** est une méthode non supervisée utilisée pour détecter les anomalies ou valeurs aberrantes dans un ensemble de données. Il repose sur l'idée que les points normaux se trouvent dans des régions denses, tandis que les anomalies sont isolées ou situées dans des zones moins denses. Pour chaque point, LOF mesure la densité locale en comparant la distance entre ce point et ses k plus proches voisins. Il calcule ensuite un score qui indique à quel point la densité autour de ce point est inférieure à celle de ses voisins. Si un point a une densité beaucoup plus faible que celle de ses voisins, son score LOF sera élevé, ce qui signifie qu'il est probablement une anomalie. Cet algorithme est très utile dans des domaines comme la

détection de fraudes, les intrusions réseaux ou les défauts industriels, car il permet d'identifier des comportements anormaux même dans des ensembles de données complexes.

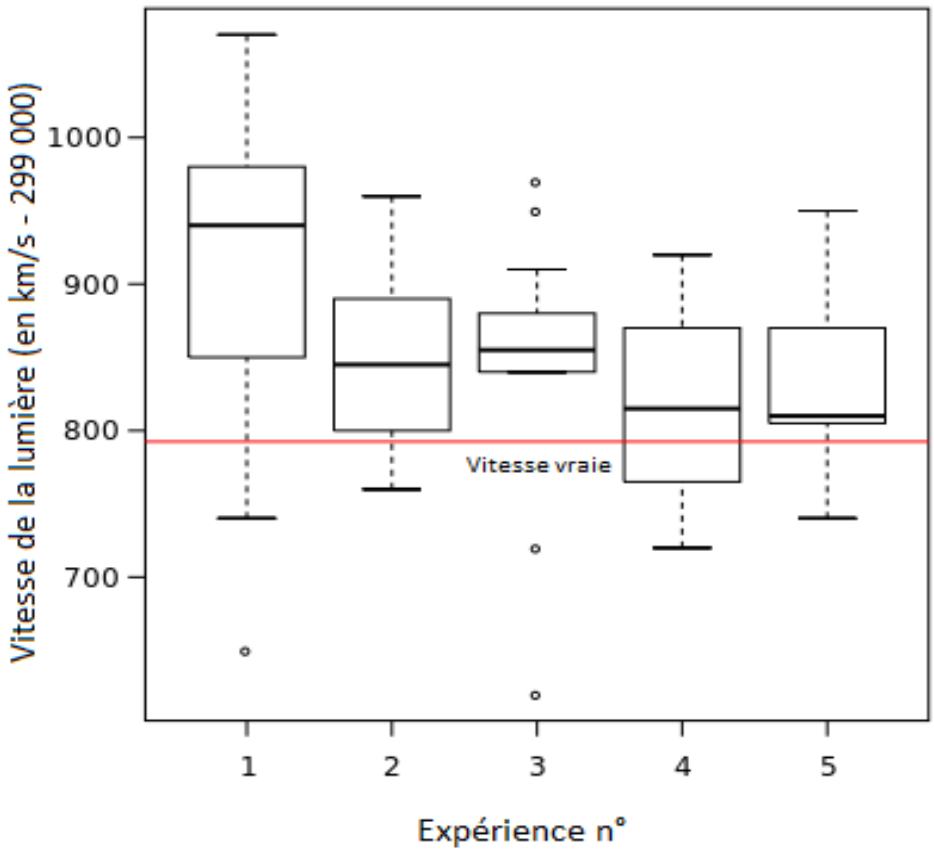

Figure 3.6 : Mesures expérimentales de la vitesse de la lumière

- **XGboost :** est un algorithme d'apprentissage supervisé très puissant, basé sur la technique du boosting**,** qui combine plusieurs arbres de décision faibles pour former un modèle robuste et performant. Le principe du boosting est d'entraîner les arbres l'un après l'autre, où chaque nouvel arbre essaie de corriger les erreurs commises par les arbres précédents. XGBoost améliore cette méthode en utilisant une approche mathématique optimisée appelée gradient boosting, où les erreurs sont corrigées en suivant le gradient de la

fonction de perte. Il est également très rapide grâce à des optimisations comme le parallélisme, la régularisation pour éviter le surapprentissage (overfitting), et la gestion automatique des valeurs manquantes. XGBoost est largement utilisé dans les compétitions de science des données et pour des tâches complexes de classification ou de régression, car il combine précision, rapidité et robustesse [11].

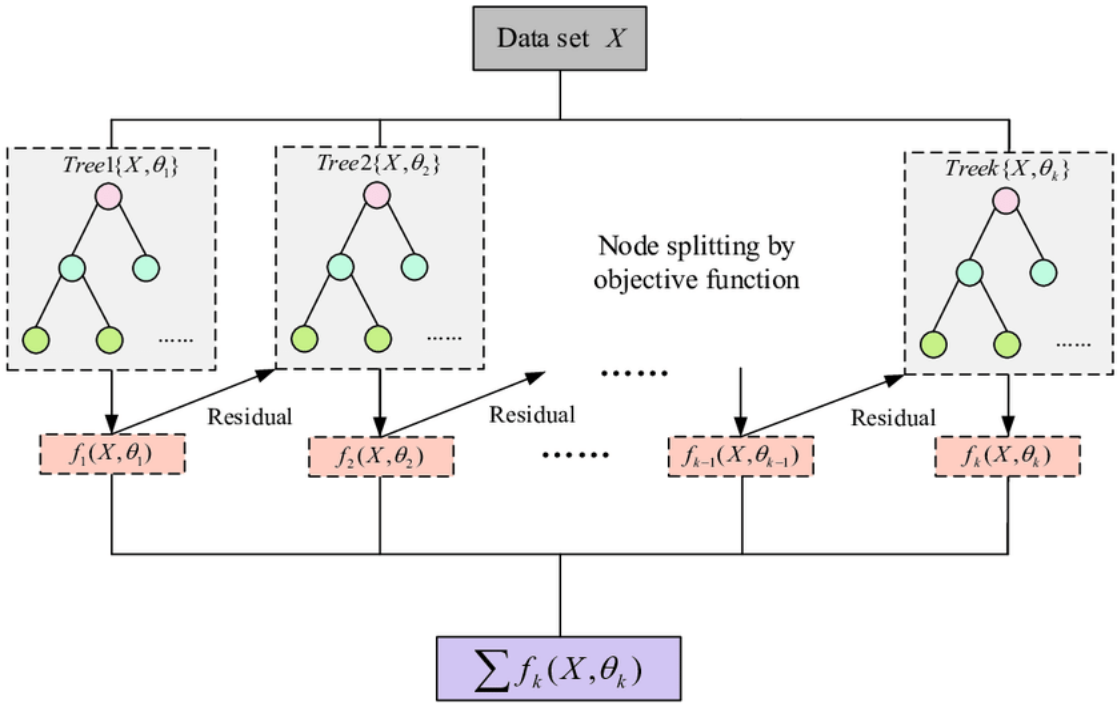

Figure 3.7 : Architecture du Gradient Boosting

3.4 APPRENTISSAGE PROFOND

Le Deep Learning est une sous-catégorie de l'apprentissage automatique qui utilise des réseaux de neurones artificiels profonds pour modéliser des relations complexes au sein des données. L'apprentissage profond nécessite généralement une phase initiale de supervision humaine pour entraîner le modèle. Une fois cette étape accomplie, le système devient capable d'apprendre et de détecter des schémas anormaux ou des attaques de manière autonome, avec une intervention humaine

minimale. Cette approche s'avère particulièrement efficace dans des environnements complexes et dynamiques comme les réseaux IoT.

3.5 RESEAUX DE NEURONES ARTIFICIELS

Les modèles de réseaux de neurones artificiels, inspirés du processus connectif du cerveau humain (biologiquement désigné comme les connexions neuronales), donnent aux ordinateurs la capacité de résoudre des problèmes de manière autonome et d'améliorer leurs aptitudes globalement. Ils sont constitués d'au moins deux couches de neurones, connues sous le nom de « Couches Cachées », comme illustré à la figure 3.8. Chaque couche renferme un grand nombre de neurones artificiels spécialisés.

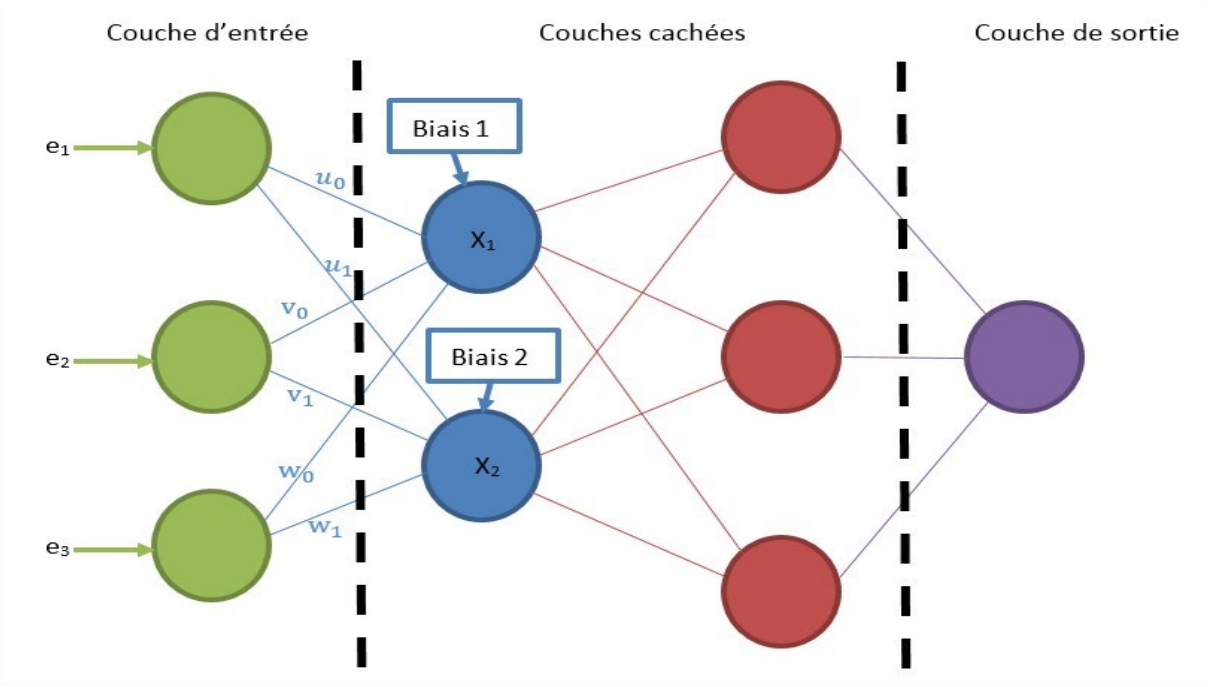

Figure 3.8 : Les couches de neurones artificiels

3.6 MISE EN PLACE DU MODELE D'APPRENTISSAGE PROFOND

Les modèles de deep learning sont formés grâce à un procédé nommé récupération. Au cours de la phase d'entraînement, on expose le modèle à des données d'entrée et on confronte ses prédictions aux valeurs cibles préétablies. Dans un réseau de neurones, le processus d'information suit une séquence définie : les données vont de la couche d'entrée vers la couche de sortie, en traversant les couches intermédiaires cachées. Chaque neurone d'une couche reçoit une valeur d'entrée qui correspond au produit de la valeur d'entrée multipliée par le poids de la connexion liée aux neurones de la couche antérieure, comme démontré dans la figure 3.1. Par la suite, à chaque niveau, une pondération est effectuée et le biais est intégré. Finalement, on utilise la fonction d'activation sur cette valeur pour déterminer les informations qui seront envoyées aux neurones suivants.

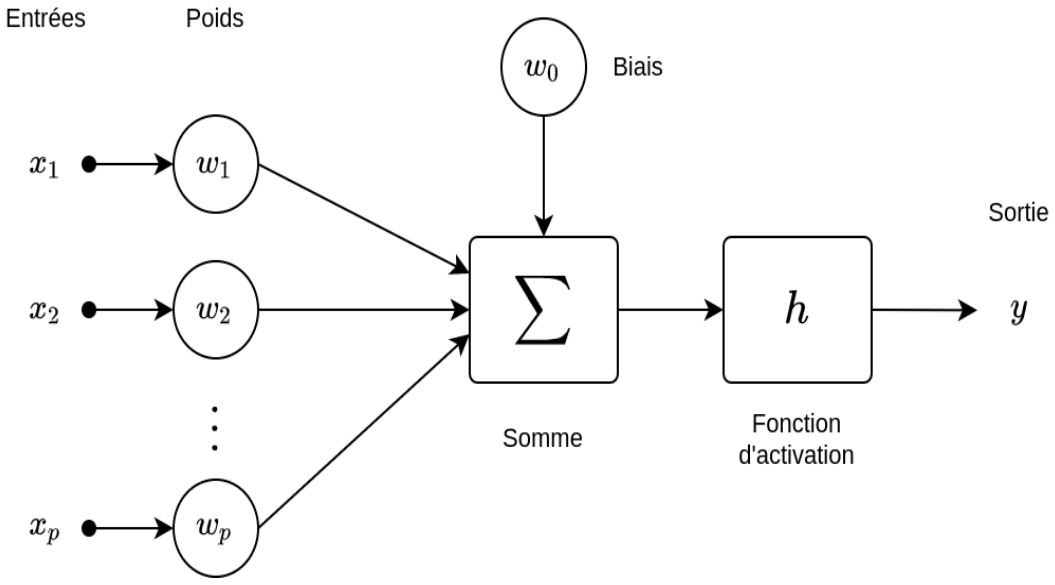

Figure 3.9 : Mécanisme de fonctionnement d'apprentissage profond

L'erreur est ajustée à travers le réseau, en modifiant les poids et biais des neurones de chaque couche pour minimiser l'erreur. Ce processus itératif se poursuit jusqu'à ce que le modèle atteigne un niveau de performance satisfaisant.

- **Couche cachée :** Dans le cadre des réseaux de neurones artificiels, une couche intermédiaire désigne une couche de neurones qui n'est ni la couche d'entrée xi, ni la couche de sortie.

- **Fonctions d'activation h :** Les fonctions d'activation injectent des non-linéarités dans le réseau neuronal, ce qui lui confère la capacité de modéliser des relations complexes. Des fonctions d'activation usuelles comprennent la sigmoïde, le tanh et le ReLU (Rectified Linear Unit). Elles sont caractérisée par:

$$y = h(\Sigma(wi \cdot xi) + b)$$

xi : Composante i de l'entrée

wi : Mesure l'importance de l'entrée xi

b : Biais

h : fonction d'activation

y : Sortie du neurone

- **Le poids (wi) :** La première opération effectuée par un neurone artificiel consiste à déterminer la somme pondérée de ses entrées. Les poids sont des valeurs numériques qui déterminent l'influence de chaque entrée sur la sortie du neurone.

- **Le Bais :** Dans les réseaux de neurones, la fonction d'activation reçoit une entrée 'x' qui est multipliée par un poids 'w'. Le biais permet de décaler la fonction d'activation en incluant une constante (à savoir le biais attribué) à l'entrée. Dans le contexte des réseaux de neurones, le biais peut être vu comme un élément équivalent à une constante dans une fonction linéaire permettant d'ajuster le seuil d'activation du neurone et d'augmenter la flexibilité du modèle.

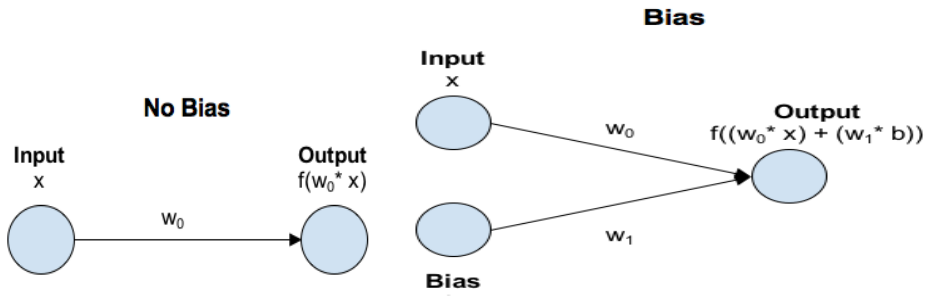

Figure 3.10 : Modèles avec et sans biais dans les réseaux neuronaux

- Deux situations sont présentées à partir de la figure 3.10 :

  — **Dans un contexte libre de biais**, l'entrée de la fonction d'activation est simplement x multiplié par le poids de connexion w_0 soit $x.w_0$.

  — **Dans un contexte avec biais,** l'entrée de la fonction d'activation devient $x.w0 + b.w1$, où b représente le biais et w_1 le poids associé à ce biais. Cela entraîne un décalage de la fonction d'activation d'une valeur fixe $(b * w1)$.

- **Seuil :** Dans certaines fonctions d'activation, le seuil est un paramètre utilisé pour décider si un neurone doit être activé (ou déclenché) ou pas. Dans les premiers modèles de réseaux de neurones, on utilisait une fonction de seuil stricte où la réponse du neurone était binaire, en fonction de si l'entrée dépassait ou non une valeur seuil prédéfinie.

$$y = 1 si\ \Sigma(wi \cdot xi)\ \geq seuil\ y = \ 0 si\ \Sigma(wi \cdot xi)\ < seuil$$

Les réseaux de neurones artificiels sont capables d'examiner les informations relatives au trafic réseau, comme les paquets IP, les adresses, les temps de réception, et autres caractéristiques, afin de distinguer les comportements normaux de ceux qui sont anormaux. Des algorithmes d'apprentissage automatique, tels que les K plus proches voisins (K-NN), le Naive Bayes et les réseaux de neurones artificiels à perceptron multicouche (DNN) peuvent être utilisés à cette fin.

3.7 ALGORITHMES DE DEEP LEARNING

Différents types de réseaux de neurones sont exploitables pour cette tâche, en fonction des propriétés des données et des buts précis. Voici quelques modèles fréquemment adoptés pour l'identification des botnets :

3.7.1 Réseaux de neurones récurrents (RNN)

Les Réseaux de Neurones Récurrents (RNN) constituent une architecture d'apprentissage profond particulièrement adaptée au traitement des données

séquentielles telles que les séries temporelles, les textes ou les flux audio. Contrairement aux réseaux neuronaux classiques, les RNN sont dotés d'une mémoire interne leur permettant de conserver une trace des états précédents de la séquence. À chaque pas de temps tt, le réseau traite une nouvelle entrée xt en la combinant avec l'état caché précédent ht−1h, produisant ainsi un nouvel état caché ht. Cette opération permet au modèle de capturer les dépendances temporelles dans les données. L'ensemble des poids du réseau est partagé à travers le temps, ce qui confère au modèle une capacité de généralisation sur des séquences de longueurs variables. Néanmoins, les RNN standards présentent certaines limitations, notamment la difficulté à modéliser des dépendances à long terme en raison du phénomène de disparition ou d'explosion du gradient. Pour pallier ces problèmes, des variantes plus robustes telles que les réseaux LSTM (Long Short-Term Memory) et GRU (Gated Recurrent Unit) ont été développées, offrant une meilleure capacité à mémoriser des informations sur des horizons temporels étendus.

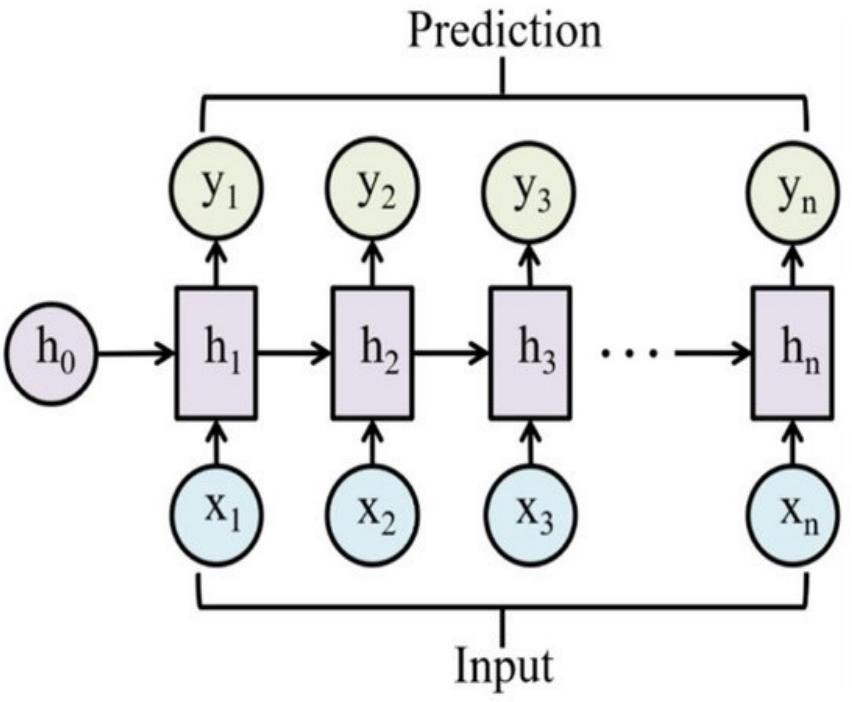

Figure 3.11 : Architecture d'un Réseau RNN

3.7.2CNN

Un réseau neuronal convolutif peut comporter des dizaines, voire des centaines de couches, chaque couche étant formée pour identifier diverses propriétés d'une image. Des filtres sont appliqués à chaque image du jeu de données d'entraînement avec diverses résolutions, puis le résultat de chaque image convoluée sert d'entrée pour la couche suivante. Initialement, ces filtres peuvent se focaliser sur des attributs très basiques tels que la lumière et les contours, mais ils évoluent progressivement pour capturer des caractéristiques typiques propres à l'objet [12].

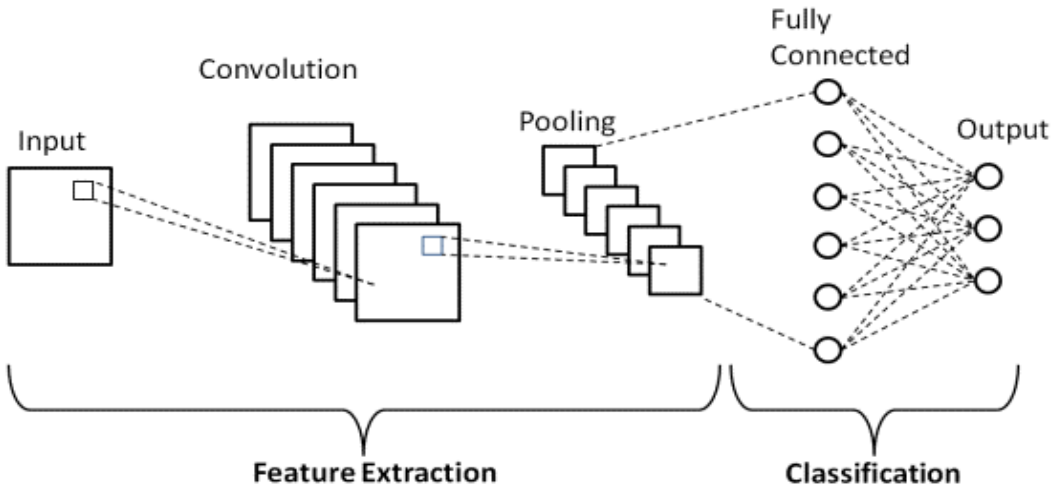

Figure 3.12 : Architecture d'un Réseau CNN

3.7.3 Réseaux (ANN) et Réseaux de neurones (DNN)

Les réseaux de neurones artificiels (ANN) sont des modèles d'apprentissage inspirés du fonctionnement biologique du cerveau humain. Ils sont composés de couches de neurones interconnectés organisés en trois types : la couche d'entrée, les couches cachées et la couche de sortie. Chaque neurone reçoit des signaux d'entrée pondérés, les additionne, applique une fonction d'activation non linéaire (telle que ReLU ou sigmoid), puis transmet le résultat aux neurones de la couche suivante.

L’apprentissage d’un ANN repose sur un processus appelé rétropropagation de l’erreur, combiné à des techniques d’optimisation comme la descente de gradient, permettant d’ajuster les poids pour minimiser l’erreur entre la sortie réelle et la sortie attendue.

Un Deep Neural Network (DNN) est une extension des ANN, caractérisée par la présence de plusieurs couches cachées, ce qui permet au modèle de capturer des représentations hiérarchiques et complexes des données. Chaque couche d’un DNN extrait des caractéristiques de plus en plus abstraites, rendant ce type de modèle particulièrement efficace pour des tâches complexes telles que la classification d’images, la reconnaissance vocale ou la détection d’anomalies dans des environnements comme l’Internet des Objets (IoT). Toutefois, l’entraînement des DNN requiert des ressources computationnelles importantes ainsi que des volumes conséquents de données pour éviter le surapprentissage et garantir une bonne généralisation.

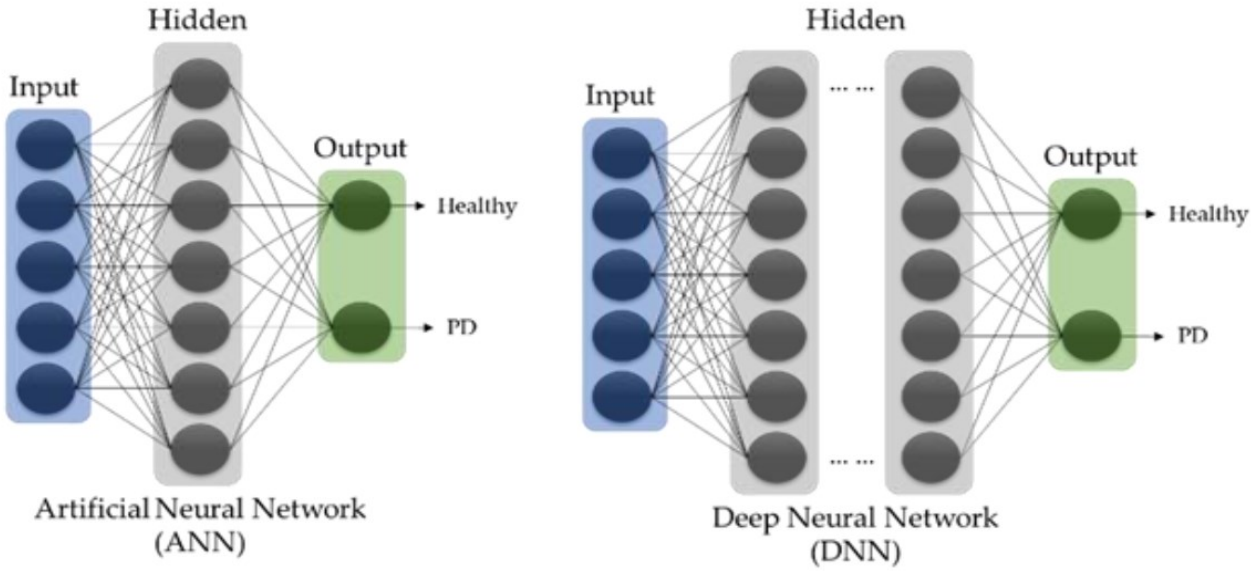

Figure 3.13 : Comparaison entre un Réseau (ANN) et un Réseau (DNN)

3.7.4 Perceptron Multicouche (MLP)

MLP est un type de réseau de neurones artificiels utilisé en apprentissage supervisé pour résoudre des problèmes de classification et de régression. Il est composé de plusieurs couches de neurones : une couche d'entrée, une ou plusieurs couches cachées, et une couche de sortie. Chaque neurone d'une couche est connecté à tous les neurones de la couche suivante à travers des poids, qui sont ajustés pendant l'apprentissage. Le fonctionnement commence par l'injection des données d'entrée (X1,X2,X3) dans la couche d'entrée. Ces données sont ensuite propagées à travers les couches cachées où chaque neurone effectue une somme pondérée des entrées qu'il reçoit, puis applique une fonction d'activation (comme ReLU ou sigmoïde) pour introduire de la non-linéarité. Le signal transformé atteint la couche de sortie, qui donne le résultat final du modèle. L'apprentissage du MLP se fait en ajustant les poids grâce à l'algorithme de rétropropagation (backpropagation), qui calcule l'erreur entre la sortie prédite et la sortie attendue, puis rétro-propage cette erreur dans le réseau pour corriger les poids. Ce processus se répète pendant plusieurs itérations (ou époques) jusqu'à ce que le modèle apprenne à prédire avec précision.

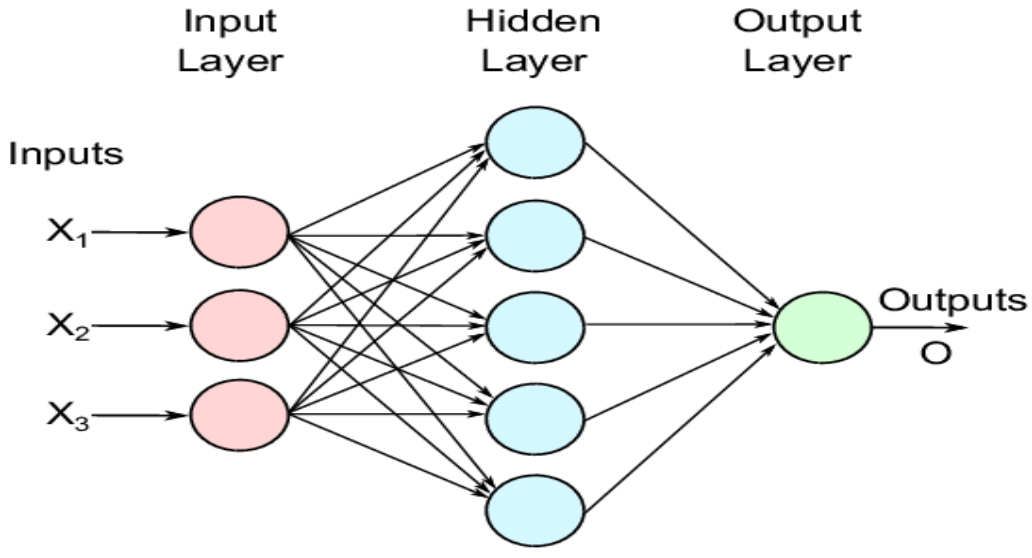

Figure 3.14 : Architecture d'un réseau de neurones (MLP)

- **Pourquoi fait-on appel aux techniques de machine learning et deep learning ?**

L'utilisation de techniques d'apprentissage automatique et profond pour détecter les attaques dans IoT a plusieurs avantages, comme l'indique le tableau ci-après.

Bénéfices	Description
Une précision renforcée	Les méthodes de ML et DL sont capables d'examiner d'importantes quantités de données et de repérer les motifs et les anomalies qui pourraient passer invisibles avec les techniques conventionnelles, ce qui conduit à une meilleure précision dans l'identification des intrusions.
Une localisation en temps réel	Les méthodes de ML et DL sont capables d'examiner les données en temps réel, facilitant ainsi une identification et une réaction plus rapide face aux attaques.
Un aspect d'évolutivité	Les méthodes de ML et DL peuvent être ajustées pour traiter d'importants volumes de données, ce qui les rend appropriées pour des contextes IoT comportant un grand nombre de dispositifs.
Diminution des résultats injustement positifs	Les méthodes de ML et DL permettent de minimiser les faux positifs, améliorant ainsi l'efficacité des ressources et diminuant le risque d'épuisement des alertes.

Tableau 3.1 : Technique de machine learning et deep learning

3.8 TYPES D'ANOMALIES

Les classifications des anomalies de données que peut identifier un système de détection d'anomalies se répartissent en deux catégories principales : involontaires et intentionnelles.

- **Anomalies involontaires :** Les irrégularités imprévues sont des points de données qui divergent de la norme en raison d'inexactitudes ou de perturbations dans le processus d'acquisition des données. Ces inexactitudes peuvent être systématiques ou stochastiques et résulter de problèmes tels que des capteurs défectueux ou des erreurs humaines lors de la saisie des données. Elles peuvent fausser l'ensemble de données, ce qui complique l'obtention d'informations précises.

- **Anomalies intentionnelles :** Ce sont des instances de données qui divergent de la norme en raison d'actions ou d'occurrences particulières. Elles peuvent fournir des informations importantes sur l'ensemble du corpus de données car elles peuvent mettre en évidence des phénomènes ou des modèles distinctifs[15].

3.9 METHODES DE DETECTION D'ANOMALIES

L'utilisation d'un cadre d'identification des aberrations pour identifier les irrégularités dans les données constitue un élément essentiel de l'étude des données afin de garantir la précision et la fiabilité des résultats. Diverses techniques d'identification des anomalies peuvent être utilisées pour construire un système.

- **Visualisation :** Est un formidable instrument pour détecter les anomalies dans les données en raison de sa capacité à faciliter l'identification rapide des valeurs aberrantes et les tendances prospectives. En représentant les données à l'aide de tableaux et de graphiques, on peut examiner visuellement l'ensemble de données pour détecter des points de données ou des tendances atypiques [13].
- **Tests statistiques :** Peuvent être utilisé pour discerner les anomalies dans les ensembles de données en comparant les données observées avec la distribution ou le modèle prévu.

  Par exemple, le test de Grubbs permet d'identifier les valeurs aberrantes dans un ensemble de données en comparant chaque point de données individuel à la moyenne et à l'écart type de l'ensemble de données. De même, le test de Kolmogorov-Smirnov est utilisé pour évaluer si un ensemble de données suit une distribution particulière, telle qu'une distribution normale [13].

3.10 TECHNIQUES DE DETECTION DES ATTAQUES

- **Détection d'attaques non supervisée :** Dans le contexte des réseaux IoT, cette méthode consiste à entraîner un modèle à partir de données non labellisées, permettant au système d'apprendre de manière autonome les comportements "normaux" et de détecter toute déviation pouvant indiquer une attaque. Bien que largement utilisée pour sa capacité à découvrir des menaces inconnues, cette approche nécessite généralement de vastes ensembles de données et une puissance de calcul importante. Elle est particulièrement adoptée dans les systèmes d'apprentissage profond, notamment avec les réseaux de neurones, pour identifier des schémas d'attaques complexes sans supervision explicite.

- **Détection supervisée :** font appel à un algorithme entraîné sur un jeu de données classées qui inclut aussi bien des cas standards que des cas déviants. En raison de l'insuffisance globale de données d'entraînement identifiées et du déséquilibre intrinsèque des classes, ces techniques de détection des anomalies sont rarement appliquées.

Semi-supervisées : Exploitez les attributs avantageux de la détection d'anomalies non supervisée et supervisée. En fournissant une fraction des données étiquetées à un algorithme, il est possible de procéder à un apprentissage partiel. Par la suite, les ingénieurs des données utilisent l'algorithme partiellement entraîné pour étiqueter de manière autonome un ensemble de données plus vaste, une méthode appelée « pseudo-étiquetage ». Si ces points de données récemment découverts se révèlent fiables, ils sont incorporés à l'ensemble de données original afin d'améliorer l'algorithme.

3.11 CONCLUSION

L'utilisation de l'intelligence artificiel présente un avantage majeur pour la détection des attaques dans l'IoT comparé aux techniques conventionnelles, grâce à sa capacité d'apprendre et de généraliser à partir de grandes quantités de données diverses, identifiant ainsi des comportements anormaux délicats et évolutifs sans avoir besoin de directives prédéfinies. À l'opposé des méthodes qui se basent sur des signes ou des règles.

Les modèles basés sur l'intelligence artificielle possèdent la capacité de s'adapter et de se développer continuellement. Toutefois, l'efficacité de l'apprentissage automatique repose sur l'accès à des bases de données d'excellente qualité et une puissance de calcul importante.

La conception d'un modèle de détection des attaques permet d'explorer différentes méthodes et algorithmes de ML et DL, facilitant ainsi l'identification de l'approche la plus adaptée à chaque contexte spécifique. Cette démarche est particulièrement cruciale pour la détection d'attaques dans les réseaux IoT, où la solution optimale dépend des caractéristiques propres à chaque type d'attaque. Ce concept sera développé dans le chapitre suivant.

CHAPITRE 4

CONCEPTION ET REALISATION

4.1 INTRODUCTION

Dernièrement, de nombreuses études ont montré l'efficacité de l'utilisation du machine learning et du deep learning pour détecter les attaques par DDoS, Brute Force , Mirai , Spoofing , Dos , WebBased , Reconnaissance qui se sont intensifiées au fil des années. Des études se concentrent aussi sur la détermination des traits ou attributs principaux (features en anglais) d'un trafic DDoS, dans le but de le différencier d'un trafic standard. Ces attributs diffèrent en fonction des jeux de données, du fait du processus d'assemblage des données brutes qui requiert un tri et une mise en forme pour l'entraînement et la validation des modèles de classification. Dans cette initiative, nous avons pour objectif de former et d'évaluer nos modèles basés sur l'apprentissage automatique et l'apprentissage profond afin de repérer les attaques par intrusion ou l'activité DDoS. Par la suite, nous allons les évaluer dans le but de déterminer la solution la plus adaptée à la détection des attaques DDoS , Brute Force , Mirai , Spoofing , Dos , WebBased , Reconnaissance dans le cadre de l'internet des objets.

4.2 OUTILS ET ENVIRONNEMENT

4.2.1 Environnement

Kaggle est une communauté virtuelle de data scientists et d'ingénieurs en apprentissage machine, qui est une filiale de Google. Kaggle est un site web réunissant la plus importante communauté de science de données à l'échelle mondiale, Il offre également des outils et des ressources puissants pour favoriser les avancées en science des données. Il facilite la recherche d'ensembles de données nécessaires à la création de modèles de l'intelligence artificielle, la publication de ces ensembles, la collaboration avec d'autres spécialistes en science des données et

ingénieurs en apprentissage automatique, ainsi que la participation à des concours visant à relever les défis du domaine.

Jupyter Notebook est un outil de création de notebooks, faisant partie du projet Jupyter. Grâce à la richesse des notebooks interactifs, Jupyter Notebook propose de nouvelles approches rapides et dynamiques pour étudier et clarifier votre code, explorer et représenter vos données, ainsi que partager vos concepts. Les notebooks font évoluer l'approche traditionnelle de l'informatique interactive vers un nouveau niveau de qualité, en offrant une application web qui permet de capturer tout le processus de calcul : la création, la documentation et l'exécution du code, ainsi que la présentation des résultats [14].

4.2.2 OUTILS UTILISES

- **Pandas :** une bibliothèque logicielle open-source, est spécifiquement élaborée pour le traitement et l'analyse de données en Python. Elle se distingue par sa puissance, sa flexibilité et sa simplicité d'utilisation.
  En fait, le terme « Pandas » est une abréviation de « Panel Data », qui désigne des ensembles de données contenant des observations sur plusieurs périodes. Cette bibliothèque a été conçue comme un instrument avancé de l'analyse en Python. Pandas permet désormais d'exploiter le langage Python pour le chargement, l'alignement, la manipulation et la fusion de données. Ses performances sont particulièrement notables quand le code est rédigé en Python ou en R.

- **Scikit-learn :** Souvent abrégé en sklearn, est une bibliothèque Python open source qui fournit des ressources performantes et accessibles pour

l'apprentissage automatique et l'analyse de données. Elle couvre divers domaines tels que la classification, la régression, le regroupement (clustering), la réduction de dimensions, le choix de modèles et la préparation des données. Elle est très répandue dans la communauté de l'apprentissage automatique grâce à sa simplicité d'utilisation, sa documentation riche et bien structurée, ainsi que la diversité des algorithmes qu'elle intègre.

- **Seaborn :** est une bibliothèque Python dédiée à la visualisation de données, construite sur la base de matplotlib. Elle fournit une interface de haut niveau pour la création de graphiques statistiques captivants et instructifs.

  Seaborn facilite l'exploration et la compréhension des données grâce à des fonctions intégrées pour représenter les distributions, les relations entre les variables et les structures de données complexes. Pour une introduction rapide aux concepts de la bibliothèque, il est possible de consulter la documentation officielle, les tutoriels ou articles de présentation. L'installation du package et les instructions d'utilisation sont disponibles sur la page d'installation du site officiel.

- **NumPy :** Abréviation de « Numerical Python », est une bibliothèque open source écrite en Python. Elle est surtout employée dans le domaine de la programmation scientifique, notamment dans les secteurs de la Data Science, de l'ingénierie, des mathématiques et des sciences en général. La Data Science exige des calculs scientifiques d'une grande complexité. Afin de mener à bien ces calculs, les Data Scientists nécessitent des outils performants. NumPy est un outil essentiel pour réaliser des opérations mathématiques et statistiques avec Python. Elle est particulièrement performante pour le traitement de

tableaux (arrays) et de matrices multidimensionnelles, offrant une grande vitesse d'exécution et une gestion efficace de la mémoire.

4.3 METHODOLOGIE D'IMPLEMENTATION DU MODELE

Dans le cadre du développement d'un modèle d'apprentissage automatique, plusieurs étapes essentielles doivent être suivies pour garantir la qualité et la performance du modèle final. Ce processus débute par l'importation et l'analyse préliminaire des données, se poursuit par un prétraitement rigoureux visant à préparer les données pour l'apprentissage, et se conclut par la conception, l'entraînement et l'évaluation du modèle à l'aide de différents algorithmes. Chaque phase joue un rôle fondamental dans la construction d'un système fiable et efficace, capable de produire des résultats pertinents à partir de données brutes.

- **La première phase** se compose trois étapes : Tout d'abord, elle consiste à importer les données depuis le site Kaggle, qui constitue une source fréquente pour les jeux de données en apprentissage automatique. Ces données sont généralement fournies au format CSV, largement utilisé dans ce domaine. Pour cette opération, nous avons recours à des bibliothèques fondamentales telles que Pandas, qui facilitent la lecture et la manipulation des fichiers. Ensuite, il est nécessaire de charger les bibliothèques indispensables au traitement et à l'analyse des données. Pandas et NumPy sont parmi les plus utilisées dans ce contexte, car elles offrent des outils performants pour la gestion de structures de données et les opérations mathématiques.Enfin, la dernière étape de cette phase consiste à effectuer une analyse préliminaire des données. Une fois les bibliothèques importées et les données chargées, il s'agit

d'en examiner les dimensions, les types de variables, ainsi que la structure générale de l'ensemble de données. Des statistiques descriptives telles que la moyenne (AVG), la variance, ou encore l'écart-type (std) sont alors calculées pour obtenir une première compréhension des caractéristiques globales du jeu de données.

- **La seconde phase** concerne le prétraitement des données, une étape cruciale qui suit la compréhension initiale du jeu de données. Elle vise à préparer les données afin de garantir la qualité et la pertinence de l'apprentissage. Ce processus inclut plusieurs opérations fondamentales telles que le traitement des valeurs manquantes, l'élimination des données nulles et la transformation des types de données si nécessaire.

  Parmi les techniques couramment utilisées, on retrouve la binarisation, qui convertit des valeurs catégorielles en format binaire et la normalisation, qui permet de mettre les données sur une échelle comparable, essentielle notamment pour les algorithmes sensibles aux écarts de valeurs.

  Il est important de souligner que les caractéristiques (attributs) des données utilisées pour entraîner un modèle d'apprentissage automatique ont un impact direct sur sa performance. Des attributs non pertinents ou partiellement informatifs peuvent non seulement réduire la précision du modèle, mais aussi augmenter la complexité computationnelle. Ainsi, un prétraitement rigoureux constitue une étape déterminante pour garantir l'efficacité du modèle final.

- **Troisième phase** pour le développement d'un modèle d'apprentissage automatique. Cette phase consiste à concevoir et entraîner un modèle en

séparant les données en deux ensembles distincts : un ensemble d'entraînement (*train*) et un ensemble de test (*test*). Cette séparation est essentielle pour garantir une évaluation impartiale des performances du modèle.

Une fois cette division effectuée, différents algorithmes d'apprentissage automatique peuvent être appliqués à l'ensemble d'entraînement. L'objectif est ensuite de mesurer et comparer l'efficacité et la précision de chaque modèle sur l'ensemble de test, afin de déterminer celui qui offre les meilleures performances dans le contexte étudié.

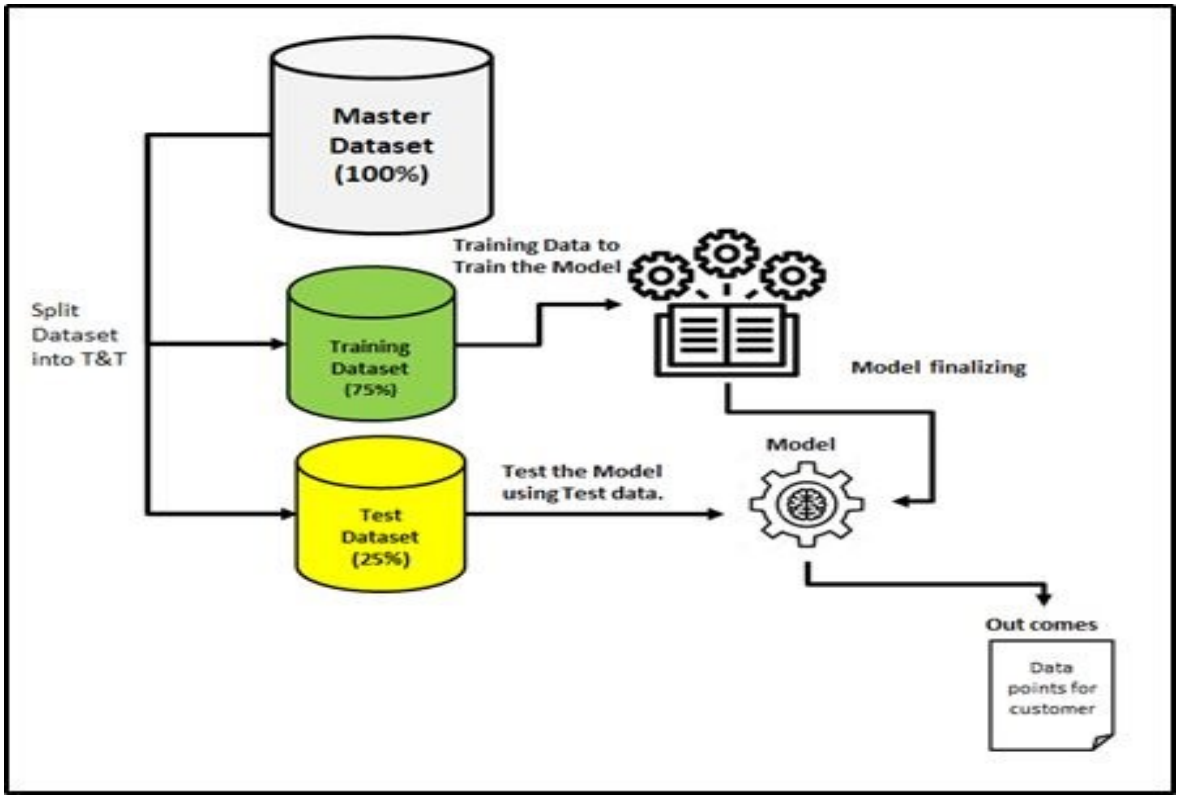

Figure 4.1: Méthodologie générale d'implémentation du modèle

4.4 METRIQUES D'EVALUATION

Pour évaluer les performances des techniques de machine learning et de deep learning, on utilise un ensemble varié de métriques. Le choix du modèle le plus adapté repose sur l'analyse de ces indicateurs.

• **La précision (accuracy) :** C'est un indicateur pertinent lorsque le jeu de données est équilibré. Dans des contextes de réseau réels, les échantillons normaux sont souvent bien plus fréquents que les échantillons anormaux. De ce fait, la précision pourrait ne pas être un indicateur pertinent.

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN}$$

TP : True Positive **FP :** False Positive

TN : True Negative **FN :** False Negative

• Précision (P) : C'est le rapport entre le nombre d'échantillons positifs réels et le nombre d'échantillons positifs prédits, illustrant ainsi la fiabilité de la détection des attaques.

$$P = \frac{TP}{TP+FP}$$

TP : True Positive

FP : False Positive

• **Loss (L) :** La fonction de perte, aussi désignée sous le nom de fonction d'erreur, constitue un composant crucial dans le domaine de l'apprentissage automatique. Elle mesure la différence entre les résultats prévus par un algorithme et les valeurs réelles visées. Par exemple, dans un cas de régression dont l'objectif est d'estimer le prix d'un véhicule en se basant sur des données passées, une fonction de perte examine une estimation faite par un réseau neuronal à partir d'un échantillon d'apprentissage tiré du ensemble de données destinées à l'entraînement. La fonction de perte mesure la différence, ou l'inexactitude, entre le coût d'une voiture estimé par le réseau et le coût réel.

$$L=-i=1\sum Cyi\cdot\log(y\hat{}i)$$

C: le nombre de classes (ou catégories possibles)

yi : la vraie étiquette (valeur réelle), souvent représentée comme un vecteur one-hot

y^i : la probabilité prédite par le modèle pour la classe i . C'est souvent la sortie d'une softmax

• **Score F1 :** C'est un critère de performance utilisé dans le cadre de l'apprentissage automatique pour juger l'efficacité d'un modèle classificateur sur un jeu de données, particulièrement dans les situations où les classes sont inégales, c'est-à-dire qu'une classe est nettement plus présente qu'une autre. C'est la moyenne harmonique du rappel et de la précision, fusionnant ces deux indicateurs en un seul chiffre qui équilibre leur importance respective.

$$F = 2 * \frac{P * R}{P + R}$$

F : F-mesure / F1-score

P : Précision

R : Rappel (Recall)

4.5 EXPERIMENTATION

4.5.1 Description du Dataset

Produire des données de sécurité IoT exploitables pour des applications pratiques présente un enjeu majeur pour diverses raisons. Un des défis majeurs découle de l'existence d'un large réseau comprenant de nombreux appareils IoT physiques, qui reflètent les topologies d'applications IoT réelles.

De nombreux projets se contentent de recourir à des dispositifs simulés ou à un nombre limité de capteurs réels, en raison des coûts élevés associés, de la nécessité d'un matériel réseau spécialisé (comme des routeurs, commutateurs et capteurs), ainsi que du personnel qualifié requis pour la maintenance de cette infrastructure.

Dans ce contexte, L'Institut canadien pour la cybersécurité (ICC) bénéficie donc d'une position notable dans l'univers de la cybersécurité, avec un passé riche en contributions importantes tant pour le secteur industriel que pour le monde académique. On peut mentionner, entre autres, les jeux de données employés pour concevoir de nouvelles applications en cybersécurité et plusieurs collaborations avec le secteur industriel afin d'optimiser les méthodes de cybersécurité et élaborer de nouvelles résolutions.

Cette réussite a donné l'opportunité au CIC d'établir un laboratoire IoT avec un réseau exclusif pour l'élaboration de solutions de sécurité IoT. En diffusant les informations acquises à partir de cette large topologie riche en données réelles. La base de données fournie par le CIC constitue un fondement essentiel pour le développement, l'évaluation et l'optimisation de solutions de sécurité dans le domaine de la cybersécurité des objets connectés, tout en soutenant de nombreuses initiatives de recherche appliquée.

4.5.2 Description des attaques

1. DDos

- **UDP flood :** Une inondation UDP est une forme d'attaque par déni de service où une multitude de paquets UDP sont acheminés vers un serveur cible, dans l'intention d'entraver sa capacité de traitement et sa réactivité. L'inondation UDP pourrait également épuiser le pare-feu qui protège le serveur ciblé, provoquant ainsi un déni de service pour le trafic légitime. [15]

[15] https://www.cloudflare.com/fr-fr/learning/ddos/udp-flood-ddos-attack/

- **SYN : Il s'agit d'un** genre d'attaque par déni de service (DDoS) qui cherche à rendre un serveur inaccessible pour le trafic authentique en épuisant toutes les ressources disponibles du serveur. En envoyant massivement des paquets de requête de connexion initiale (SYN), l'attaquant parvient à saturer tous les ports disponibles sur un serveur ciblé, ce qui oblige l'équipement ciblé à réagir lentement face au trafic authentique, ou même à l'empêcher complètement.

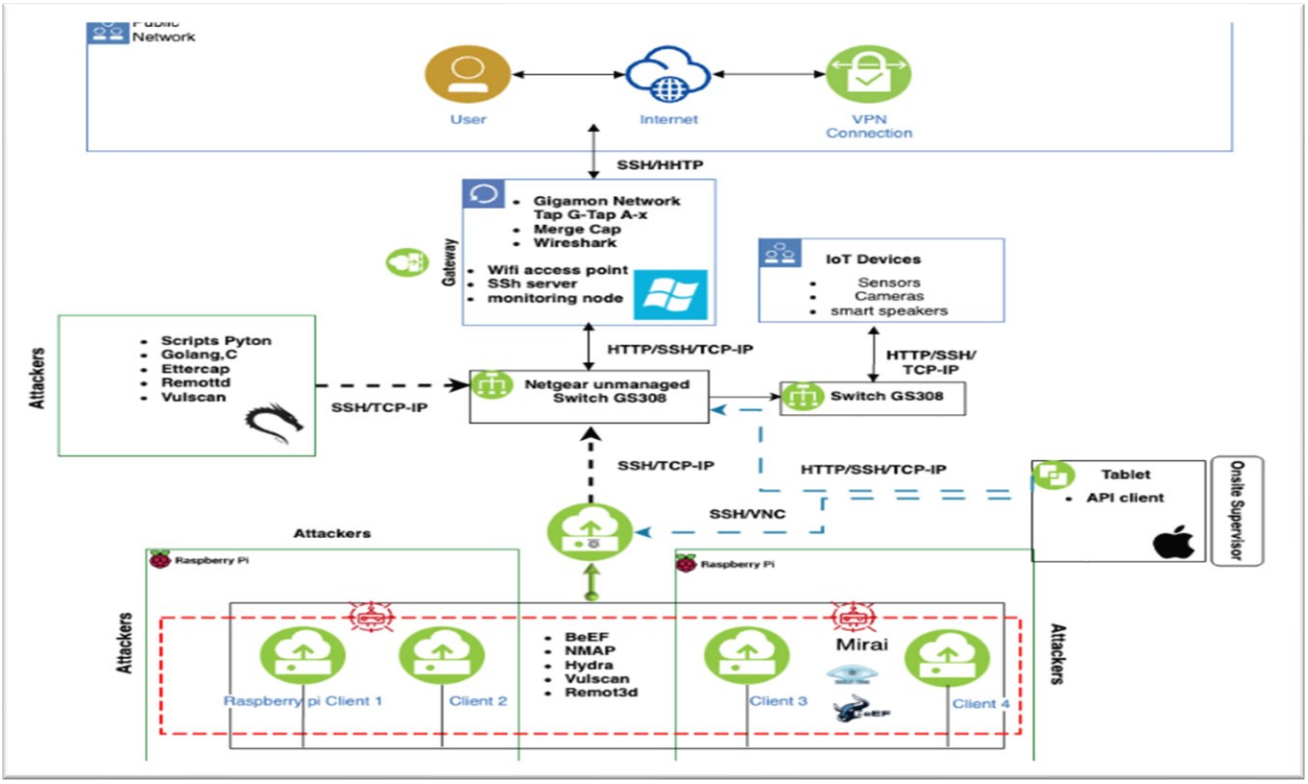

Figure 4.2 : Topologie d'un Réseau IoT avec scénarios d'attaques et de surveillance

- **ICMP :** Une attaque de Ping flood, également connue comme une attaque par déni de service, consiste à saturer un appareil cible avec des paquets de requête d'écho ICMP afin de le rendre inaccessible aux flux de trafic légitime. Quand l'agression vient de nombreux appareils, il s'agit alors d'une attaque DDoS ou déni de service distribué [16].

2. Brute force

C'est une technique de piratage qui consiste à tester et devenir des mots de passe, des identifiants d'accès et des clés cryptographiques. Il s'agit d'une stratégie simple mais efficace pour obtenir sans permission l'accès à des comptes personnels, ainsi

qu'aux réseaux et systèmes d'organisations. L'assaillant tente différentes combinaisons de noms d'utilisateur et de mots de passe, généralement avec l'aide d'un ordinateur, afin de découvrir les informations d'accès pertinentes [17].

3 .Mirai

C’est un malware qui vise les dispositifs intelligents basés sur des processeurs ARC, les convertissant en une botnet ou un réseau de « zombies » pilotés à distance. On utilise fréquemment ce réseau de bots, connu sous le nom de botnet, pour mener des attaques DDoS [18].

4 . Dos

Une attaque par déni de service (DoS) est un type de cyberattaque dans lequel un acteur malveillant vise à rendre un ordinateur ou un autre appareil indisponible pour ses utilisateurs prévus en interrompant le fonctionnement normal de l'appareil. Les attaques DoS fonctionnent généralement en submergeant ou en saturant une machine ciblée de requêtes jusqu'à ce que le trafic normal ne puisse plus être traité, ce qui entraîne un déni de service pour les utilisateurs supplémentaires. Une attaque DoS se caractérise par l'utilisation d'un seul ordinateur pour lancer l'attaque [19].

5 . Spoofing

Est une cybermenace où un hackeur se fait passer pour une entité valide en falsifiant des informations comme une adresse e-mail, un numéro de téléphone ou une adresse IP. Le but est de tromper la victime pour obtenir des informations confidentielles ou infiltrer un réseau. Contrairement au phishing, qui repose sur des

messages trompeurs, il agit d'un niveau plus technique, simulant directement l'identité d'une personne ou d'une organisation.[20]

[20] https://www.mailinblack.com/ressources/glossaire/quest-ce-que-le-spoofing/

6 . Reconnaissance

Des pirates externes peuvent utiliser des outils Internet, comme les utilitaires nslookup et whois, pour découvrir facilement les adresses IP attribuées à une entreprise ou à une entité donnée. Une fois ces adresses IP connues, l'assaillant peut lancer des requêtes ping vers les adresses publiquement accessibles pour déterminer celles qui sont actives. Pour automatiser cette étape, l'assaillant peut utiliser un outil de balayage comme fping ou gping, qui envoie systématiquement des requêtes ping à une plage d'adresses ou à toutes les adresses d'un sous-réseau [21].

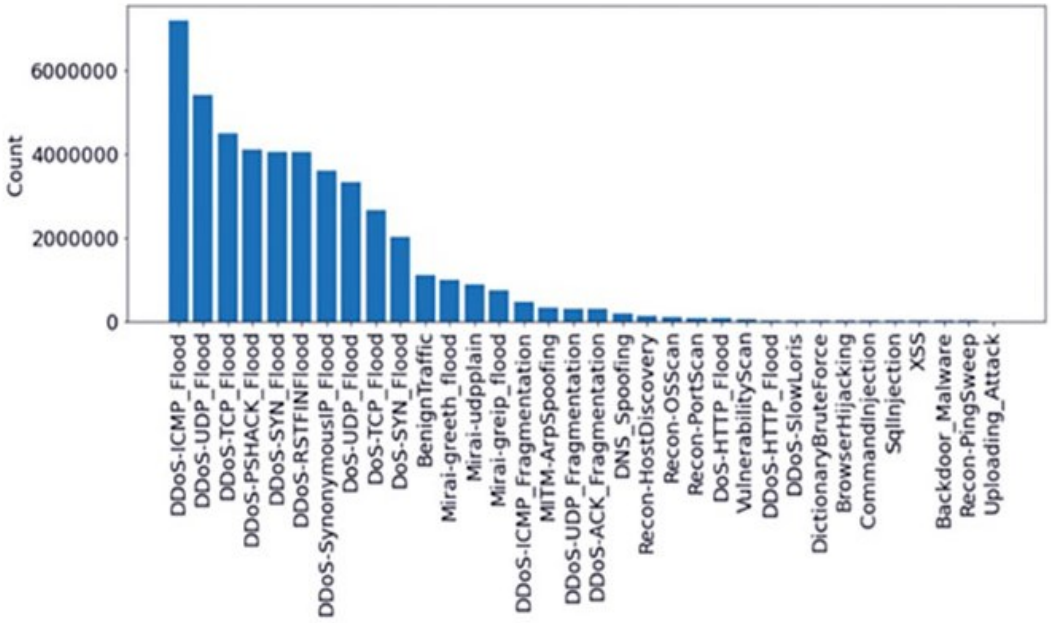

Figure 4.3 : Différents types D'attaques

La figure 4.3 illustre la fréquence des différentes sortes d'attaques informatiques détectées dans un ensemble de données. On observe que les attaques par déni de service distribué (DDoS), telles que DDoS-ICMP Flood, DDoS-UDP Flood et DDoS-TCP Flood, se produisent de manière généralisée, avec des millions d'anomalies. On retrouve également d'autres formes d'attaques, tels que les attaques par logiciels malveillants (Malware), les portes dérobées (Backdoor) ou les injections SQL, mais en quantités plus réduites. Cela démontre que les pirates privilégient les attaques DDoS, probablement pour saturer les serveurs et rendre les services indisponibles. Pour faire simple, ce graphique souligne l'importance des attaques DDoS par rapport aux autres types de cyberattaques [22].

4.6 REPERTOIRES DE JEU DE DONNEES

Le dossier principal du jeu de données (CICIoT2023) abrite quatre sous-dossiers associés à divers fichiers, à savoir :

— **PCAP :** Contient le trafic capturé pendant les attaques, enregistré sous forme de fichiers .pcap.

— **CSV :** Regroupe les attributs extraits des fichiers .pcap, utilisés pour l'évaluation par apprentissage automatique (fichiers .csv).

— **Documents additionnels :** Code source et description des outils utilisés pour le processus complet de collecte et de traitement des données liées aux attaques.

Mergecap est utilisé pour fusionner plusieurs fichiers .pcap, PySpark pour le traitement des données, et TCPDump pour segmenter les fichiers.

Le découpage des fichiers .pcap en segments plus petits a été effectué, et les caractéristiques ont été extraites à l'aide de DPKT [23].

Feature	mean	std	min	25%	50%	75%	max
flow_duration	5.76544939	285.034171	0	0	0	0.10513809	394357.207
Header_Length	76705.9637	461331.747	0	54	54	280.555	990714775
Protocol type	9.06568989	8.94553292	0	6	6	14.33	47
Duration	66.3507169	14.0191881	0	64	64	64	255
Rate	9064.05724	99562.4906	0	2.09185589	15.7542308	117.384754	8388608
Srate	9064.05724	99562.4906	0	2.09185589	15.7542308	117.384754	8388608
Drate	5.46E-06	0.00725077	0	0	0	0	29.7152249
fin_flag_number	0.08657207	0.28120696	0	0	0	0	1

Figure 4.4 : Statistiques Descriptives des caractéristiques du trafic Réseau

4.7 IMPORTATION DE LIBRAIRES

- ✓ Utilisation de NumPy pour effectuer des calculs numériques.
- ✓ Pandas pour le traitement l'analyse et la manipulation des données.
- ✓ Seaborn pour la représentation visuelle des données Python, construit sur matplotlib.
- ✓ Matplotlib propose une interface comparable à Matlab pour la réalisation de graphiques.

Les lignes suivantes importent differentes classes et fonctions spécifiques de sklearn utilisées pour la création et l'évaluation de modèles d'apprentissage automatique tels que les classificateurs SVM, RandomForest et Naive Bayes.

```
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, confusion_matrix, accuracy_score
from sklearn.svm import SVC
from sklearn.metrics import classification_report
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import classification_report, accuracy_score
from sklearn.impute import SimpleImputer
```

Les lignes suivantes importent diverses classes et fonctions spécifiques à la création et à l'évaluation de modèles d'apprentissage profond, tels que les algorithmes MLP, DNN et CNN, implémentés à l'aide de la bibliothèque TensorFlow.

```
import tensorflow as tf
from sklearn.neural_network import MLPClassifier
from sklearn.metrics import classification_report, confusion_matrix, accuracy_score
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, LabelEncoder
from sklearn.metrics import classification_report, confusion_matrix
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Dropout
from tensorflow.keras.layers import Dense, Dropout, BatchNormalization
from tensorflow.keras.callbacks import EarlyStopping, ReduceLROnPlateau
```

Ces importations concernent des bibliothèques et modules destinés à la préparation des données, à l'élaboration de modèles d'apprentissage automatique ainsi qu'à l'évaluation de leurs performances. Le module de prétraitement de sklearn fournit des outils pour préparer les données avant l'entraînement, tels que la standardisation et la normalisation. On utilise la fonction train_test_split de

sklearn.model_selection pour segmenter les données en groupes de traitement et de test.

Les classificateurs SVC et MLP utilise respectivement les classificateurs de sklearn.svm et sklearn.neural_network pour effectuer la classification SVM et la classification par réseaux de neurones multicouches. TensorFlow est une bibliothèque d'apprentissage automatique employée pour la création et l'entraînement de modèles basés sur des réseaux de neurones profonds. On utilise les modules de tensorflow.keras comme Sequential, Dense et EarlyStopping pour établir la structure des modèles et superviser leur entraînement. En définitive, on utilise to_categorical de tensorflow.keras.utils pour transformer des labels en format numérique catégoriel.

4.8 NORMALISATION ET STANDARDISATION

La première étape de la normalisation de cette base de données consiste à transformer tous les attributs de type objet en nombres entiers : (Protocole Type ,Timetolive , Flag, Rate, Label , Header_Length , Variance, AVG).

La seconde consiste à standardiser les attributs : en soustrayant la moyenne des données X et en divisant par l'écart-type pour chaque colonne, on parvient à atténuer l'influence des différentes échelles entre les colonnes sur les résultats de l'apprentissage automatique. Par la suite, 30% des données seront destinées à constituer le jeu de test du modèle d'apprentissage, tandis que les 70% restants serviront à l'entraînement.

4.9 ANALYSE STATISTIQUE

Pour mieux comprendre la répartition des différents types d'attaques dans le jeu de données consolidé après regroupement, nous effectuons une visualisation des attributs clés. L'examen des histogrammes permet d'identifier la fréquence et la distribution des attaques présentes. **La Figure 4.5** illustre clairement cette répartition.

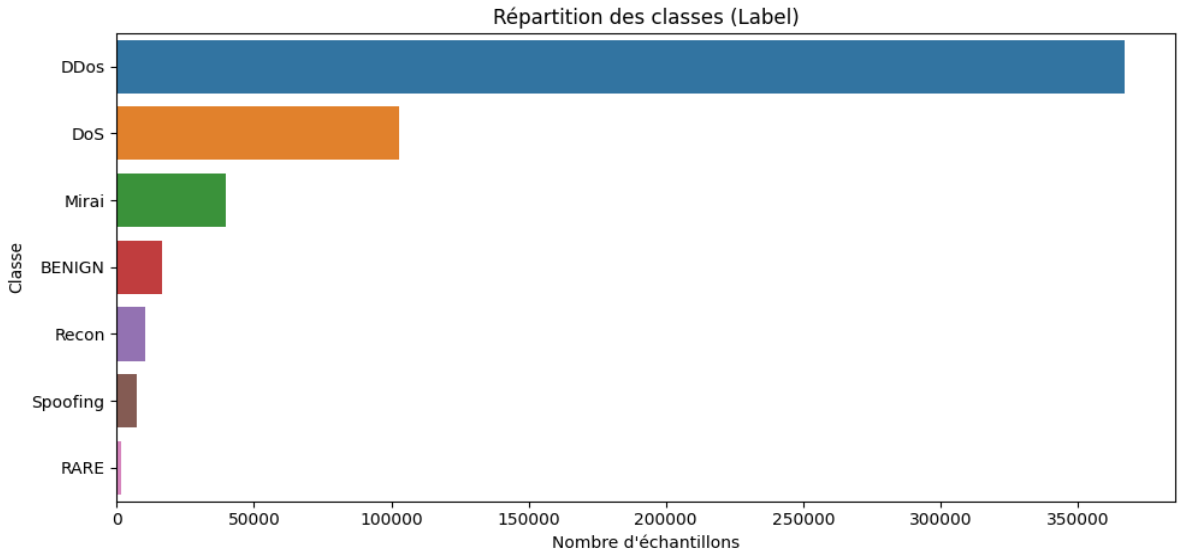

Figure 4.5 : Visualisation des types d'attaques dans le jeu de données consolidé après le regroupement

La figure 4.6 montre une matrice de corrélation entre diverses caractéristiques numériques d'un jeu de données, probablement provenant d'un trafic réseau informatique. On utilise fréquemment cette représentation pour comprendre les liens linéaires entre les variables, élément essentiel lors des phases de prétraitement et d'analyse exploratoire.

La matrice s'appuie sur une gamme de couleurs allant du bleu (indiquant une corrélation négative) au rouge sombre (signe d'une corrélation positive), où des

teintes plus claires reflètent des corrélations faibles ou absentes. La diagonale principale reste constamment en rouge sombre (valeur 1), car chaque variable entretient une corrélation parfaite avec sa propre présence.

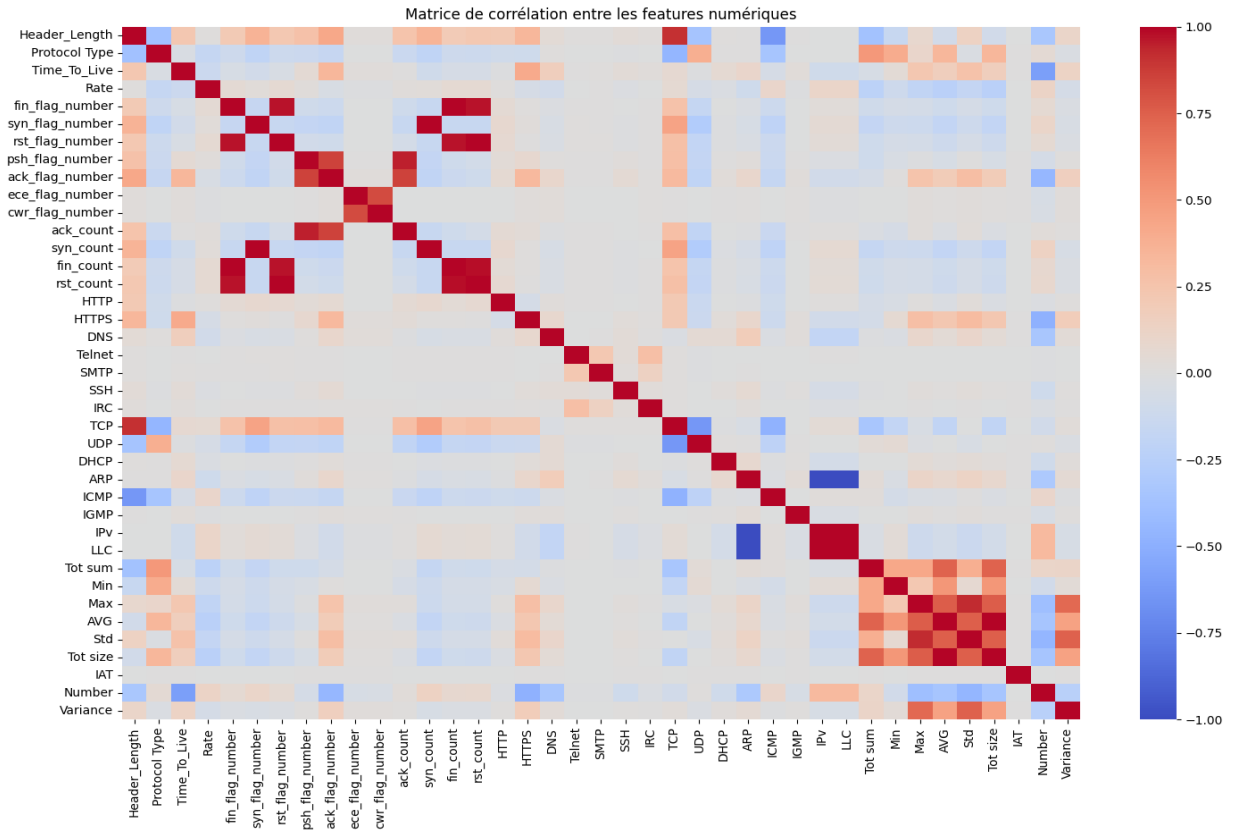

Figure 4.6 : Matrice de corrélations pour l'exploration des relations entre variables

L'analyse de cette matrice de corrélation met en lumière des relations intéressantes entre les variables :

— **Fortes corrélations positives** On remarque entre les variables associées aux flags TCP (nombre de flags SYN, nombre de flags ACK, nombre de drapeaux FIN) une forte corrélation mutuelle. Cela pourrait signifier qu'elles se

retrouvent fréquemment simultanément dans certaines catégories de flux de réseau.

On note également des corrélations fortes entre les statistiques globales telles que Total sum, AVG, Max, Std et Total size, ce qui est prévisible puisqu'elles représentent des éléments similaires de la charge réseau.

— **Corrélations négatives** Des corrélations inverses sont observées entre certains indicateurs ou protocoles et des variables comme ICMP et UDP. Cela peut démontrer des divergences essentielles dans le comportement des protocoles (par exemple : ICMP ne fait pas appel aux indicateurs TCP).

— **Corrélations faibles ou nulles** Une majorité des cases de la matrice se trouvent près du blanc/gris clair, signifiant qu'il n'y a pas de corrélation notable entre diverses variables. Cela implique que ces variables peuvent renfermer informations supplémentaires non redondantes, utiles pour l'étude ou la modélisation.

4.10 EVALUATION ET DISCUSSION DES RESULTATS

4.10.1 Modèle de Machine learning

Suite à l'entraînement de nos modèles, notamment le RandomForestClassifier, le SVM linéaire, la régression logistique, l'XGboost et l'évaluation de leur précision, exactitude, rappel et score F1, nous présentons les résultats obtenus dans le tableau ci-joint.

ALGORITHME		Précision	Recall	F1-score	support	ACCURACY
LOGISTIC REGRESSION	0	0.78	0.56	0.65	4969	0.714
	1	0.94	0.66	0.78	110147	
	2	0.41	0.82	0.54	30793	
	3	1.00	0.99	0.99	11870	
	4	0.21	0.65	0.32	409	
	5	0.68	0.71	0.69	3157	
	6	0.41	0.65	0.50	2135	
RANDOM FOREST	0	0.80	0.86	0.83	4969	0.800
	1	0.85	0.88	0.86	110147	
	2	0.51	0.45	0.48	30793	
	3	1.00	1.00	1.00	11870	
	4	0.83	0.44	0.58	409	
	5	0.74	0.76	0.75	3157	
	6	0.93	0.82	0.87	2135	
XGBOOST	0	0.80	0.87	0.84	4969	0.843
	1	0.84	0.96	0.90	110147	
	2	0.74	0.36	0.49	30793	
	3	1.00	1.00	1.00	11870	
	4	0.81	0.45	0.58	409	
	5	0.75	0.75	0.75	3157	
	6	0.95	0.82	0.88	2135	
SVM LINAIRE	0	0.69	0.62	0.65	4969	0.788
	1	0.84	0.91	0.88	110147	
	2	0.55	0.36	0.44	30793	
	3	1.00	0.99	0.99	11870	
	4	0.07	0.63	0.13	409	
	5	0.72	0.44	0.54	3157	
	6	0.38	0.28	0.32	2135	

Tableau 4.1 : Résultats des modèles d'apprentissage automatique

a. RandomForestCLassifier

L'algorithme de forêt aléatoire est constitué d'un ensemble d'arbres de décision, chaque arbre étant constitué d'un échantillon de données tiré d'un ensemble d'apprentissage avec remise, appelé échantillon bootstrap. Un tiers de cet échantillon d'apprentissage est réservé comme données de test, appelé échantillon out-of-bag (oob), sur lequel nous reviendrons plus tard. Un autre élément aléatoire est ensuite injecté par regroupement de caractéristiques, ce qui ajoute de la diversité à l'ensemble de données et réduit la corrélation entre les arbres de décision [24].

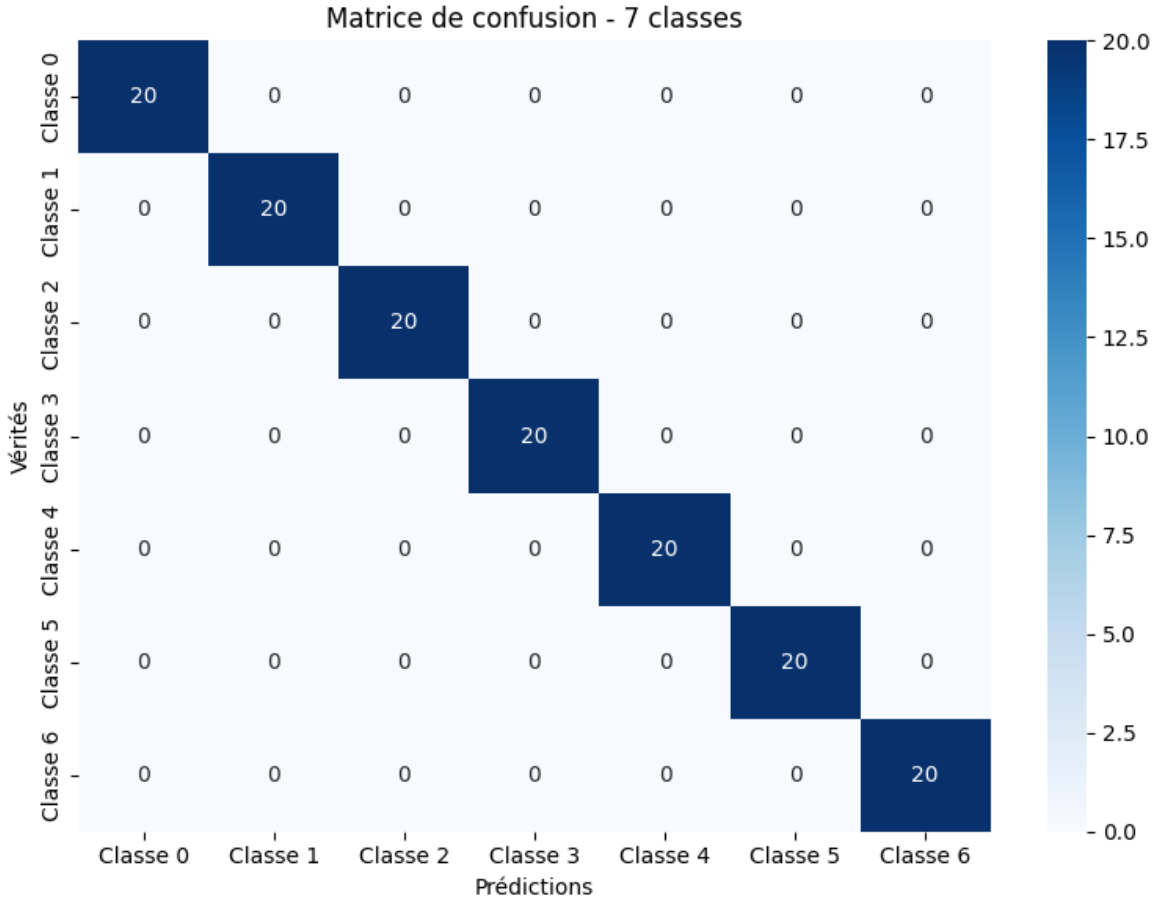

Figure 4.7 : Matrice de confusion RandomForest

La matrice de confusion présentée évalue les performances d'un modèle de classification sur un problème à 7 classes (Classe 0 à Classe 6). Elle montre que pour chaque classe, le modèle a correctement prédit 20 instances, soit un total de 140 prédictions exactes sur 140. Toutes les valeurs sont situées sur la diagonale principale, ce qui indique que le modèle n'a commis aucune erreur de classification. Aucune confusion entre les classes n'a été observée, ce qui signifie que les valeurs de rappel, de précision et de F1-score sont toutes égales à 1.0 pour chaque classe. Cette performance parfaite se traduit par une exactitude globale de 100 %. Toutefois, si cette matrice a été obtenue sur les données d'entraînement, il convient de rester prudent, car elle peut révéler un surapprentissage. En revanche, si elle résulte d'un jeu de test indépendant, elle confirme une excellente capacité de généralisation du modèle.

- **Évaluation critique du modèle d'apprentissage :**

D'après le tableau 4.1, Le modèle Random Forest montre une amélioration significative avec une accuracy de 80,0 %. Il offre une bonne précision et un bon rappel sur la majorité des classes, en particulier les classes 1, 2 et 3, où le F1-score atteint ou dépasse 0.85. La performance est également raisonnable pour les classes 4, 5 et 6, avec des F1-scores supérieurs à 0.75, ce qui indique une meilleure capacité à gérer l'équilibre entre précision et rappel sur des classes moins représentées. Dans l'ensemble, ce modèle offre des résultats solides et stables, ce qui le rend bien adapté à une classification multiclasse avec des volumes variés.

b. SVM linéaire

L'idée clé de l'algorithme SVM est de trouver l'hyperplan qui sépare le mieux deux classes en maximisant la marge entre elles. Cette marge correspond à la distance entre l'hyperplan et les points de données les plus proches (vecteurs de support) de

chaque côté. Le meilleur hyperplan, également appelé « marge dure », est celui qui maximise la distance entre l'hyperplan et les points de données les plus proches des deux classes. Cela garantit une séparation nette entre les classes.[25]

[25]https://www.geeksforgeeks.org/support-vector-machine-algorithm/

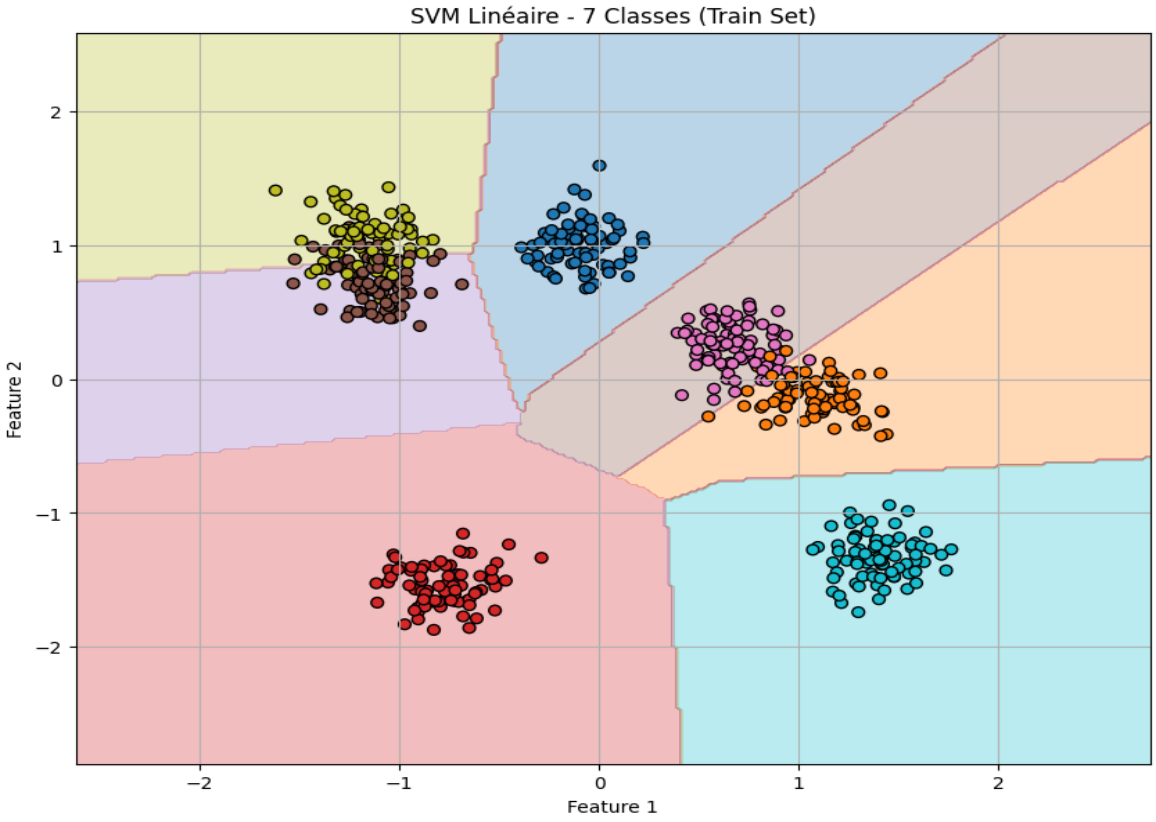

Figure 4.8 : Apprentissage Supervisé avec SVM Linéaire

La figure 4.8 représente la frontière de décision générée par un classifieur SVM linéaire appliqué à un ensemble de données d'entraînement contenant 7 classes distinctes. Chaque couleur correspond à une région de l'espace où le modèle prédit une classe particulière, tandis que les points représentent les échantillons du jeu de données, regroupés selon leurs vraies classes. On observe que les données sont globalement bien séparées, et que les frontières linéaires tracées par le modèle segmentent clairement l'espace en fonction des deux caractéristiques (Feature 1 et

Feature 2). Les groupes de points de chaque classe sont compacts, et bien localisés dans leurs régions respectives, ce qui montre que le modèle SVM parvient efficacement à distinguer entre les différentes classes. La linéarité des frontières reflète l'utilisation d'un noyau linéaire, adapté ici puisque les données semblent presque linéairement séparables. L'interprétation visuelle renforce donc la qualité du modèle, suggérant une bonne généralisation sur l'ensemble d'entraînement, avec très peu de confusion apparente entre les classes.

- **Analyse des Résultats par Catégorie :**

Le SVM linéaire affiche une accuracy de 78,8 % selon le tableau 4.1 , ce qui le positionne entre la régression logistique et la forêt aléatoire. Bien que la performance soit correcte pour la classe 1 (F1-score de 0.86) et acceptable pour la classe 3 (F1-score de 1.00), elle est nettement inférieure pour les autres classes. La classe 2, par exemple, souffre d'un faible rappel (0.39) et d'un F1-score de 0.46. Les classes 4 à 6 présentent également des F1-scores bas, révélant une difficulté du SVM linéaire à gérer les classes minoritaires ou moins linéairement séparables. Ce modèle est donc moins robuste que Random Forest ou XGBoost dans ce contexte.

c. RégressionLogistique

Le modèle de régression logistique transforme la sortie de valeur continue de la fonction de régression linéaire en sortie de valeur catégorielle à l'aide d'une fonction sigmoïde qui mappe tout ensemble de variables indépendantes à valeur réelle en entrée dans une valeur comprise entre 0 et 1. Cette fonction est connue sous le nom de fonction logistique [26].

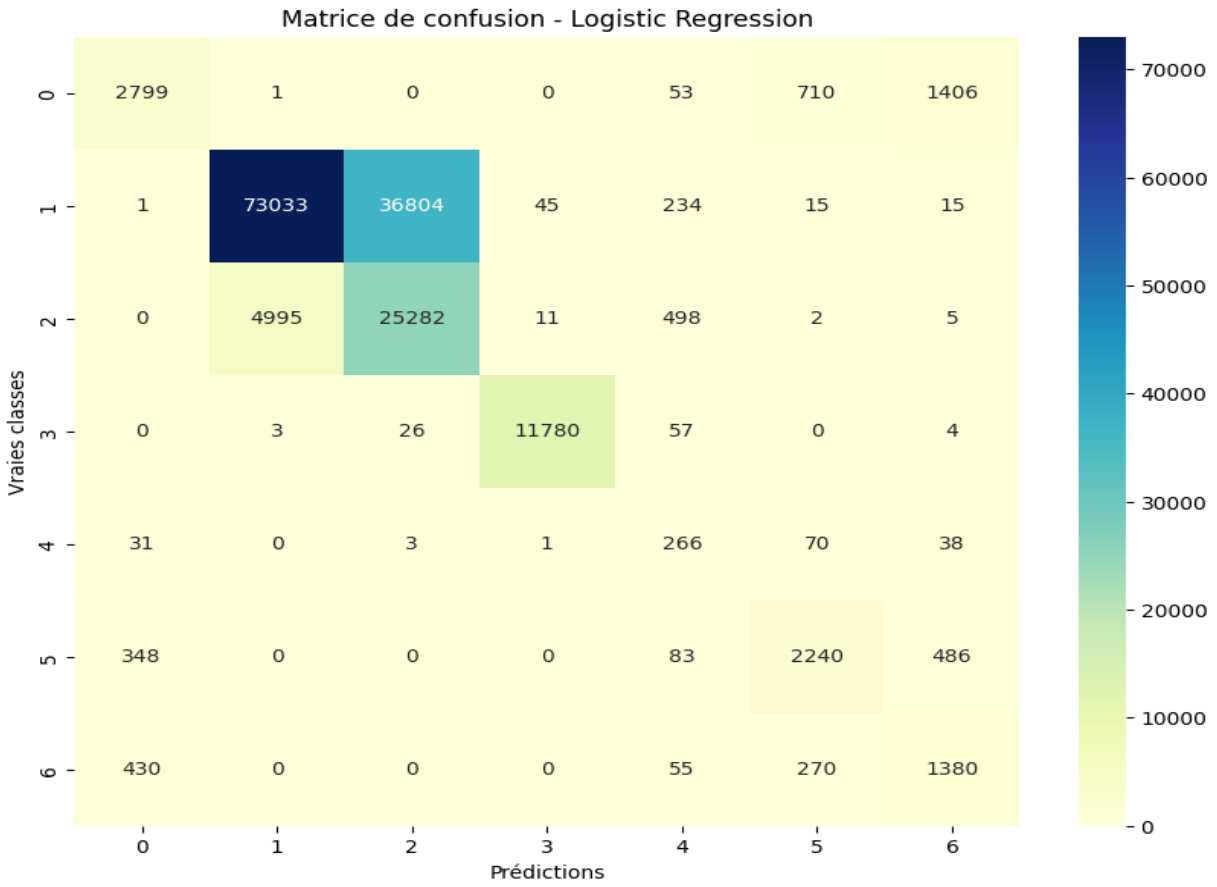

Figure 4.9 : Matrice de confusion LR

La figure 4.9 montrer une matrice de confusion issue de l'application d'un modèle de régression logistique sur un problème de classification à 7 classes (de 0 à 6). Les lignes correspondent aux classes réelles, tandis que les colonnes indiquent les prédictions faites par le modèle. Une lecture diagonale montre les prédictions correctes, tandis que les valeurs hors diagonale indiquent les erreurs de classification. On constate que le modèle obtient des résultats acceptables pour certaines classes (comme la classe 1 avec 73 033 bonnes prédictions), mais montre des taux d'erreur élevés dans d'autres cas, notamment une forte confusion entre les classes 1 et 2, où plus de 36 000 instances de la classe 1 ont été mal classées en classe 2, et près de 5 000 instances de la classe 2 ont été prédites à tort comme étant de la classe 1.

D'autres erreurs notables apparaissent dans les classes 5 et 6, avec respectivement 486 et 270 échantillons mal classés, ce qui peut impacter la précision globale du modèle. Cette matrice révèle ainsi les limites de la régression logistique dans ce contexte multiclasses complexe, notamment en présence de classes déséquilibrées ou chevauchantes, et suggère que des modèles plus puissants ou des techniques de traitement supplémentaires (comme la normalisation, la réduction de dimensionnalité ou des méthodes de rééchantillonnage) pourraient être nécessaires pour améliorer la performance globale.

- **Forces et faiblesses du modèle** :

Le tableau 4.1 montre que l'algorithme de régression logistique affiche une accuracy globale de 71,4 %, ce qui est la plus faible parmi les quatre modèles évalués. Bien que les performances soient satisfaisantes pour la classe majoritaire (classe 1) avec une précision de 0.94 et un rappel de 0.86, les résultats pour les autres classes sont très déséquilibrés. Notamment, la classe 2 présente une précision très faible (0.41) et une F1-score de seulement 0.54, révélant une forte confusion. Les classes 4, 5 et 6 sont également mal classifiées, avec des F1-scores très bas (0.32, 0.39 et 0.50 respectivement). Ce modèle semble avoir du mal à bien distinguer les classes minoritaires ou complexes, ce qui le rend peu adapté à un jeu de données déséquilibré ou complexe.

d. XGBoost (eXtreme Gradient Boosting)

Est une implémentation avancée du Gradient Boosting, une technique d'apprentissage ensembliste séquentielle qui construit un modèle prédictif en combinant itérativement une série de "faibles apprenants", typiquement des arbres de décision peu profonds, où chaque nouvel arbre est entraîné pour corriger les erreurs (résidus) des prédictions cumulées des arbres précédents. Sa force réside

dans sa fonction d'objectif qui intègre non seulement la mesure de l'erreur (fonction de perte), mais aussi un puissant terme de régularisation (L1 et L2) pour contrôler la complexité des arbres et prévenir le surapprentissage, assurant ainsi une meilleure généralisation du modèle. Lors de la construction des arbres, XGBoost utilise une approximation de second ordre pour optimiser la fonction objectif et calcule un "gain" pour chaque division de nœud, prenant en compte la réduction de la perte et la pénalité de régularisation. En plus de ces fondations mathématiques, XGBoost se distingue par des optimisations techniques majeures : il permet la parallélisation de la recherche des meilleurs points de division, gère efficacement les données creuses, peut traiter des ensembles de données hors-mémoire (out-of-core computing), et est optimisé pour l'accès à la mémoire cache, ce qui lui confère une performance et une rapidité exceptionnelles. De plus, il intègre un taux d'apprentissage (learning rate) pour stabiliser le processus et un élagage intelligent des arbres pour éviter la complexité inutile, faisant d'XGBoost un choix de prédilection pour des tâches d'apprentissage automatique de haute performance [27].

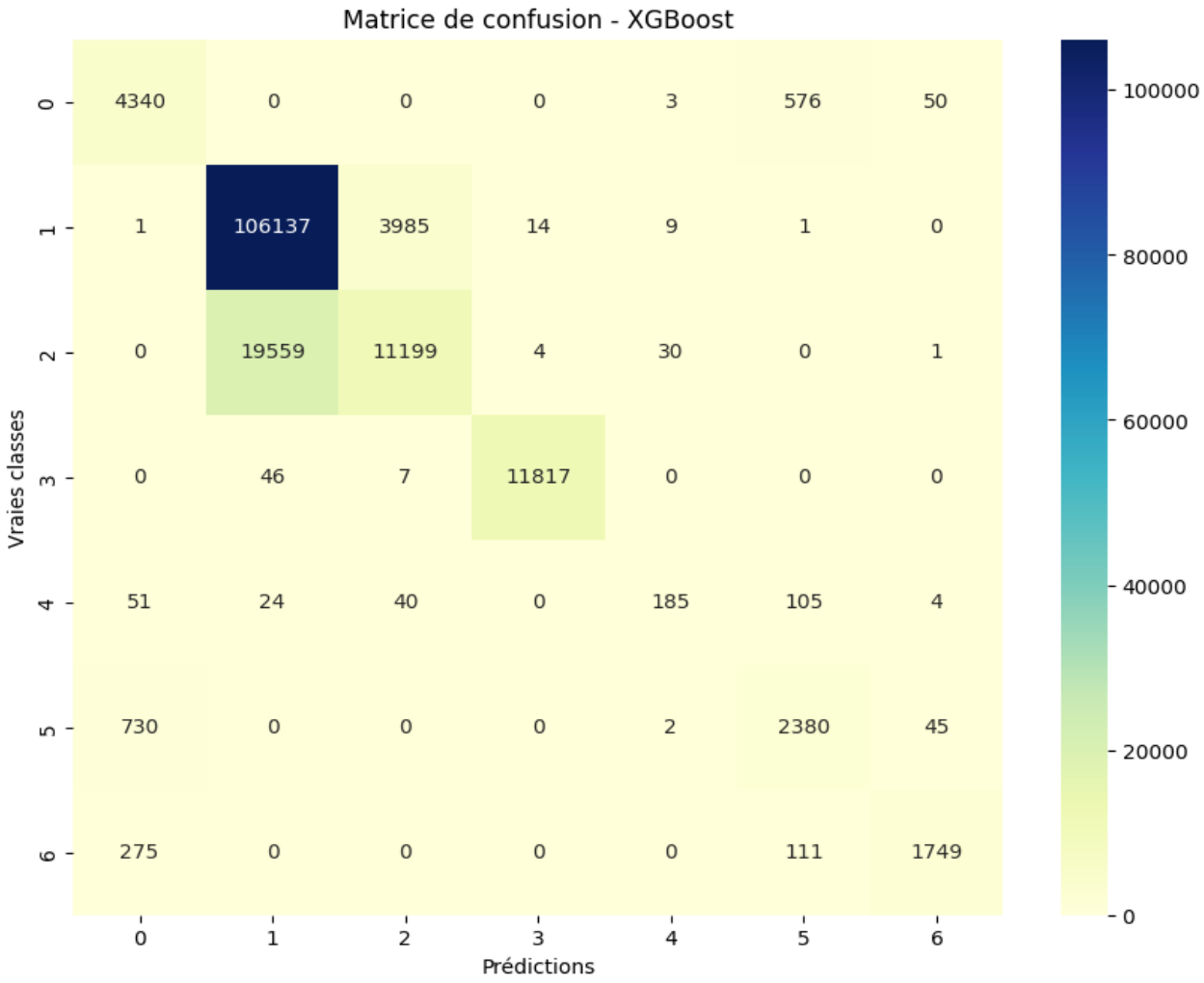

Figure 4.10 : Matrice de confusion XGBoost

La figure 4.10 présentée illustre la performance du modèle XGBoost pour une tâche de classification multiclasse comportant 7 classes (de 0 à 6). On remarque que le modèle réussit très bien à prédire la classe 1, avec 106 137 bonnes classifications contre seulement quelques erreurs mineures vers les autres classes (notamment 3985 vers la classe 2). Cependant, des confusions notables apparaissent entre certaines classes, en particulier entre les classes 1 et 2 : 19 559 exemples de la classe 2 ont été mal classés comme appartenant à la classe 1, ce qui traduit une confusion importante entre ces deux catégories. De même, la classe 0 est globalement bien prédite (4340 bonnes prédictions), mais 576 instances ont été attribuées à la classe 5, ce qui suggère

une ambiguïté modérée entre ces deux classes. Les classes 5 et 6 montrent aussi une certaine confusion mutuelle, avec 45 erreurs de la classe 5 vers la classe 6 et 111 de la classe 6 vers la classe 5. Enfin, les classes 3 et 4 semblent relativement bien distinguées, bien que la classe 4 soit plus sujette aux erreurs, notamment vers les classes voisines. En résumé, le modèle montre une bonne capacité de classification générale, mais des efforts supplémentaires sont nécessaires pour améliorer la distinction entre les classes fortement confondues, en particulier entre les classes 1 et 2.

- **Étude des écarts de performance d'algorithme :**

XGBoost est le modèle le plus performant du tableau, avec une accuracy globale de 84,3 %. Il affiche des résultats excellents pour toutes les classes, avec des F1-scores très élevés (souvent supérieurs à 0.85), et une précision et un rappel très équilibrés. Pour la classe 1 (majoritaire), la précision est de 0.94 et le rappel de 0.96. Même pour les classes complexes comme 2 et 4, le F1-score est très satisfaisant (respectivement 0.80 et 0.75). Ce modèle excelle dans la gestion des classes déséquilibrées tout en maintenant une précision globale élevée, ce qui démontre une excellente capacité d'apprentissage et de généralisation.

e. Synthèse et discussion des résultats

Le tableau 4.1 met en évidence la performance comparative de quatre algorithmes d'apprentissage automatique, à savoir la régression logistique, le Random Forest, XGBoost et le SVM linéaire, dans le contexte d'un problème de classification impliquant 7 classes. En général, XGBoost montre des performances optimales avec une précision moyenne de 84,3 %, témoignant de sa grande aptitude à gérer des données complexes et déséquilibrées. Le modèle Random Forest est à la suite avec

une précision de 80 %, démontrant une stabilité appréciable sur la plupart des classes, spécifiquement les classes majoritaires. Par contre, la régression logistique et le SVM linéaire affichent des performances plus discrètes, avec des taux de précision respectifs de 71,4 % et 78,8 %, et éprouvent des problèmes évidents concernant les classes moins fréquentes (particulièrement les classes 4, 5 et 6), comme le démontrent leurs scores F1 relativement bas. Ceci est en partie dû à la caractéristique linéaire de ces modèles, qui les rend moins performants dans la représentation de limites complexes. Ces résultats mettent en évidence l'importance de sélectionner les algorithmes favorables en fonction de la distribution des classes et de la complexité des données, et attestent de l'avantage des modèles d'ensemble tels que XGBoost dans ce cas.

4.10.2 Modèle d'apprentissage profond

ALGORITHME	Précision	Accuracy	Loss
DNN	82.86%	0.8293	0.0331
CNN	80.22%	0.8086	0.0479
MLP	93%	0.1685	1.9456

Tableau 4.2 : Résultats des modèles d'apprentissage profond

a. Évaluation quantitative du modèle DNN

D'après le tableau 4.2 le modèle DNN affiche de solides performances globales, avec une précision de 82,86 %, une accuracy de 0,8293 et une perte (loss) relativement faible de 0,0331. Ces indicateurs suggèrent que le modèle est non seulement précis, mais aussi stable et fiable sur l'ensemble du jeu de test. Le faible niveau de perte traduit un bon ajustement du modèle aux données sans surapprentissage apparent. Ce modèle semble donc bien entraîné et performant, notamment dans le cadre d'un environnement de classification multiaclasse.

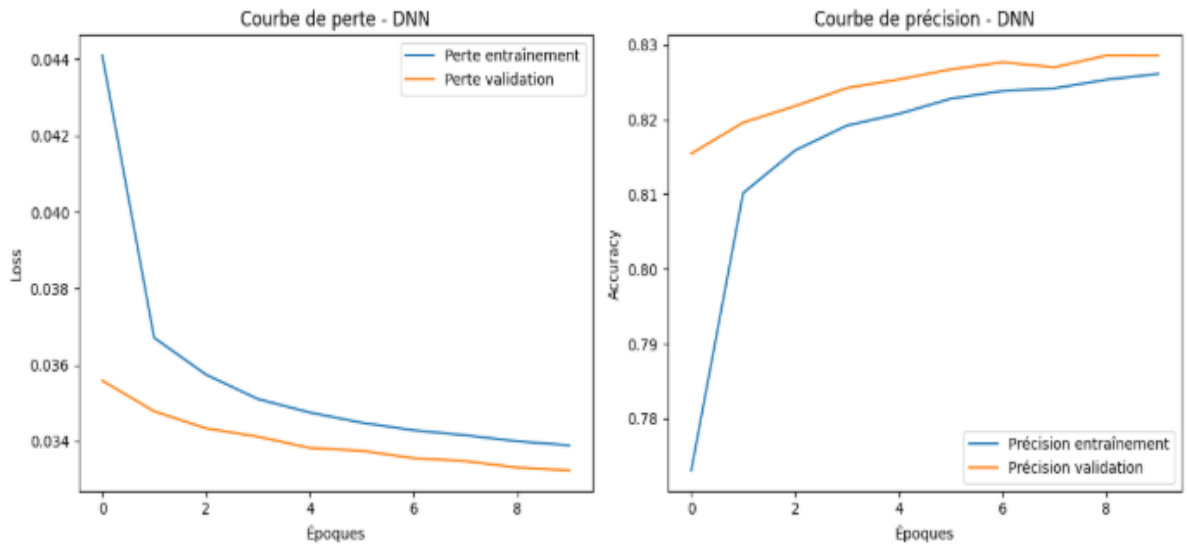

Figure 4.11 : Evolution de la perte et la précision du DNN

La figure 4.11 présente deux courbes d'apprentissage du modèle DNN (Deep Neural Network) : à gauche, la courbe de perte, et à droite, la courbe de précision, toutes deux évaluées sur les ensembles d'entraînement et de validation au fil des époques.

Sur la courbe de perte, on observe une diminution progressive et régulière de la perte (loss) aussi bien sur l'ensemble d'entraînement que de validation, ce qui indique que le modèle apprend efficacement sans signe de surapprentissage (overfitting). La perte d'entraînement passe d'environ 0.044 à 0.034, tandis que la perte de validation diminue également de façon stable pour atteindre environ 0.033 à la fin des 9 époques. Fait notable, la perte de validation est constamment inférieure à celle de l'entraînement, ce qui suggère une bonne généralisation du modèle sur des données non vues.

Quant à la courbe de précision, elle montre une augmentation continue de la précision pour les deux ensembles. La précision d'entraînement démarre à environ 0.77 et atteint près de 0.825, tandis que la précision de validation progresse légèrement au-dessus, allant de 0.815 à environ 0.83. Là encore, la précision sur les données de validation reste légèrement supérieure à celle de l'entraînement, ce qui est un signe rassurant, montrant que le modèle ne se contente pas de mémoriser les données d'apprentissage, mais généralise bien aux données nouvelles.

En résumé, cette double courbe montre que le modèle DNN est bien entraîné, avec une bonne convergence, pas de surapprentissage visible, et une généralisation efficace. Le nombre d'époques (9 ici) semble suffisant pour atteindre une performance stable, sans nécessiter un entraînement prolongé.

b. Étude de la capacité de généralisation du CNN

Dans le tableau 4.2 montre que le CNN présente une précision de 80,22 % et une accuracy de 0,8086, des performances légèrement inférieures à celles du DNN. Sa perte est de 0,0479, un peu plus élevée que celle du DNN, ce qui peut traduire une

moindre capacité d'optimisation ou une plus grande sensibilité au bruit dans les données. Cela dit, les performances restent bonnes et indiquent une capacité de généralisation correcte, notamment pour les tâches où l'extraction automatique de caractéristiques locales (via convolutions) est bénéfique.

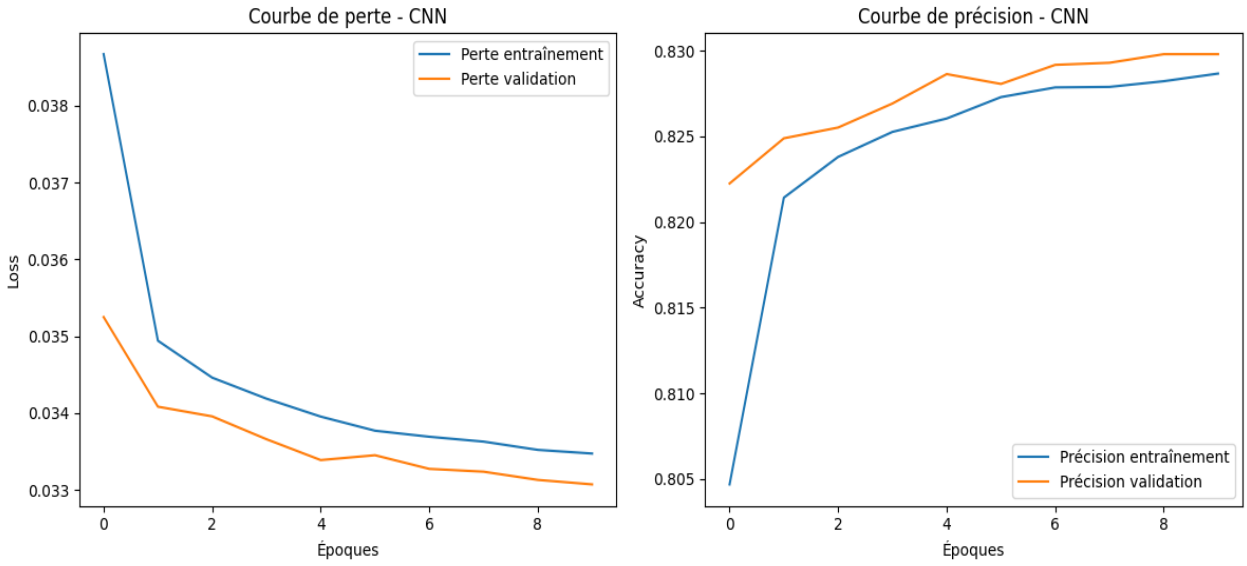

Figure 4.12 : Evolution de la perte et la précision du CNN

La figure 4.12 présente deux graphiques illustrant l'évolution de la performance d'un modèle CNN (Convolutional Neural Network) au cours de l'entraînement, sur 10 époques. Le graphique de gauche montre la courbe de perte, tandis que celui de droite illustre la courbe de précision.

Dans la courbe de perte, on observe que les pertes d'entraînement et de validation diminuent de manière régulière au fil des époques, indiquant que le modèle apprend progressivement à mieux s'ajuster aux données. Notamment, la perte de validation est légèrement inférieure à celle d'entraînement dès la première époque et continue à baisser, ce qui suggère une bonne capacité de généralisation du modèle sans

surapprentissage apparent. Les courbes suivent une trajectoire parallèle et convergente, renforçant cette impression d'un apprentissage stable et contrôlé.

Dans la courbe de précision, la précision d'entraînement commence plus bas mais augmente rapidement pour se rapprocher de celle de la validation. La précision de validation est globalement supérieure sur l'ensemble des époques, atteignant environ 83 % dès la 9e époque. Cette stabilité et cette supériorité de la courbe de validation indiquent non seulement que le modèle ne surapprend pas, mais aussi qu'il est potentiellement bien régularisé et adapté à la tâche.

Dans l'ensemble, cette figure témoigne d'un modèle CNN performant et bien entraîné, dont les performances sur les données de validation sont cohérentes et légèrement meilleures que sur les données d'entraînement, ce qui est généralement le signe d'un bon comportement de généralisation.

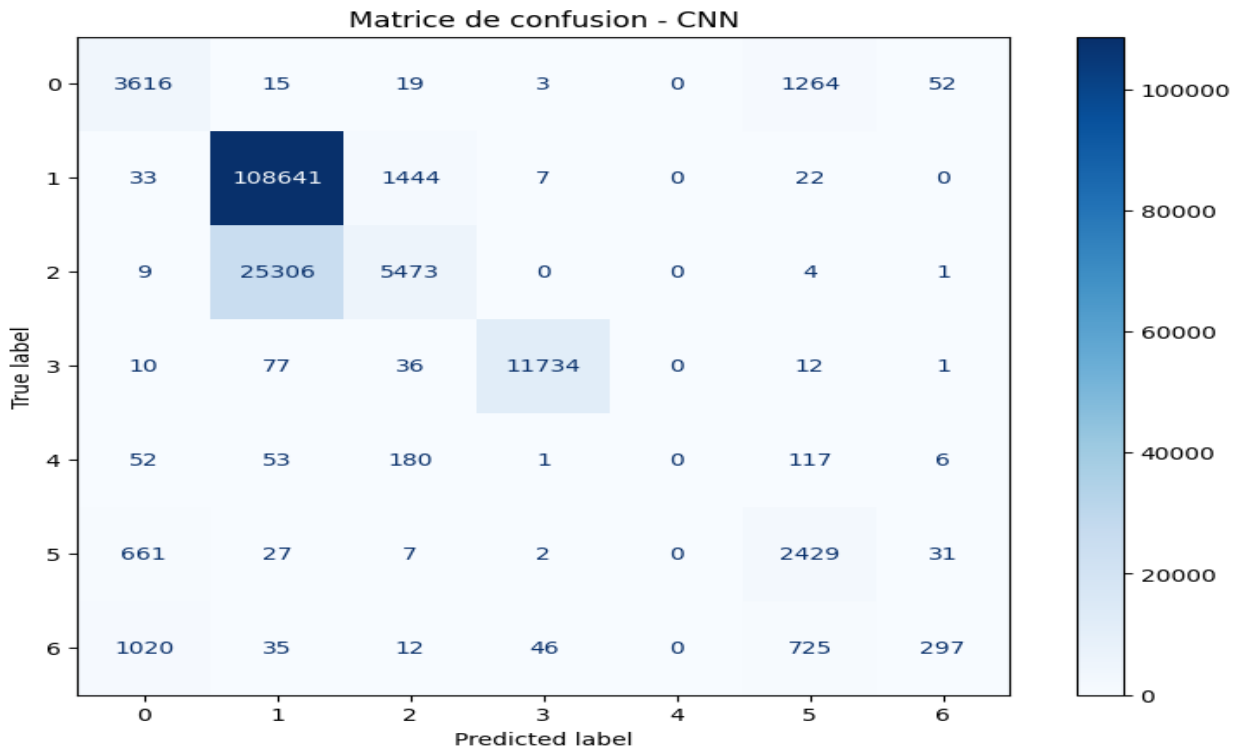

Figure 4.13 : Matrice de confusion d'un modèle CNN

La figure 4.13 montre que la matrice de confusion d'un modèle CNN appliqué à un problème de classification comportant 7 classes (étiquettes 0 à 6). Chaque cellule indique le nombre d'échantillons dont la vraie classe (axe vertical) correspond à la prédiction du modèle (axe horizontal). L'objectif d'une bonne classification est que les valeurs soient maximisées sur la diagonale principale (de haut en bas à gauche), ce qui indique des prédictions correctes.

À première vue, le modèle effectue très bien la classification pour la classe 1, avec 108 641 prédictions correctes, bien plus élevées que les erreurs, ce qui montre une très bonne sensibilité et précision pour cette classe. En revanche, la classe 2 est problématique, car une grande partie de ses instances (25 306) a été incorrectement prédite comme appartenant à la classe 1. Cela indique une confusion importante entre

les classes 1 et 2, qui pourrait s'expliquer par une forte similarité dans leurs caractéristiques ou un déséquilibre dans les données.

La classe 3 est globalement bien prédite (11 734 corrects), avec quelques erreurs dispersées. En revanche, la classe 0 subit des confusions notables avec les classes 5 (1 264 erreurs) et 6 (52 erreurs), ce qui peut réduire la précision globale de cette classe. Les classes 5 et 6 souffrent également de confusions croisées, avec par exemple 661 exemples de la classe 5 prédits comme classe 0, ou encore 1 020 exemples de la classe 6 prédits comme classe 0.

La classe 4 semble être difficile à identifier également : sur 409 exemples, seuls 117 sont correctement classés, le reste étant éparpillé dans les classes 0, 1, 2 et 5. Cela pourrait indiquer soit un nombre d'échantillons trop faible pour l'apprentissage, soit un chevauchement important avec d'autres classes.

En résumé, cette matrice met en évidence une très bonne performance globale pour certaines classes (notamment la classe 1), mais aussi des zones de confusion préoccupantes entre certaines paires de classes (surtout 1 vs 2, 0 vs 5/6), suggérant que le modèle pourrait être amélioré par des stratégies comme un meilleur équilibrage des données, un affinage des caractéristiques, ou l'utilisation de techniques de régularisation ou d'attention pour mieux discriminer les classes similaires.

c. Contribution du MLP à la classification multiclasse

Bien que le MLP affiche une précision apparemment très élevée de 93 %, cette performance est trompeuse, car l'accuracy chute drastiquement à 0,1685 et la perte est extrêmement élevée (1,9456). Cette incohérence signale probablement un problème de surapprentissage sévère, de déséquilibre dans les classes, ou encore une

mauvaise calibration des sorties. Le modèle semble bien prédire une classe majoritaire (d'où la précision élevée), mais échoue largement à généraliser sur l'ensemble des classes, comme l'indique la très faible accuracy. En l'état, ce modèle n'est pas fiable.

d. Synthèse et discussion des résultats

Le tableau 4.2 illustre une comparaison des performances entre trois modèles d'apprentissage profond : le réseau de neurones à plusieurs couches (DNN), le réseau de neurones convolutif (CNN) et le perceptron multicouche (MLP). Le modèle DNN démontre une forte performance avec un taux de précision de 82,86 %, une precision de 0,8293 et une perte minime (loss = 0,0331), ce qui témoigne d'une bonne capacité de généralisation et d'une convergence efficace. Malgré une précision légèrement inférieure (80,22 %) et une precision (0,8086), le CNN affiche aussi un faible taux de perte (0,0479), témoignant de sa fiabilité et stabilité, particulièrement pour les données dotées d'une structure spatiale.

Cependant, le MLP présente une anomalie : malgré une précision impressionnante (93 %), son taux de réussite est particulièrement bas (0,1685) et sa perte est fortement élevée (1,9456), indiquant un potentiel surapprentissage, une mauvaise évaluation du modèle ou une différence dans la mesure des performances. Cette divergence provoque des interrogations liées à la fiabilité des conclusions du MLP et exige une réflexion du modèle ou de la procédure d'évaluation. Dans l'ensemble, DNN apparaît comme le modèle le plus équilibré et performant pour cette tâche.

4.11 CONCLUSION

L'étude comparative des modèles standards d'apprentissage automatique et des modèles d'apprentissage profond démontre que les méthodes basées sur les ensembles, comme XGBoost et Random Forest, présentent d'excellentes performances, en particulier face à des classes déséquilibrées et des frontières stratégiques complexes. En ce qui concerne les modèles profonds, le DNN se distingue par sa stabilité et son aptitude à la généralisation, affichant une précision et une perte très remarquables. Cependant, les résultats du MLP indiquent des anomalies ou une sur-adaptation, ce qui nécessite un réglage ou une validation plus rigoureuse. Par conséquent, même si les modèles profonds présentent un potentiel significatif, leur performance est directement liée à leur conception et à leur paramétrage. En ce qui concerne les données structurées, des modèles typiques bien ajustés tels que XGBoost restent une référence fiable, alors que les modèles profonds comme le DNN démontrent un potentiel intéressant lorsqu'ils sont correctement formés. Ainsi, la sélection du modèle se basera sur la nature des données, les ressources à disposition et les objectifs de performance visés.

CONCLUSION GENERALE

Dans un univers technologique en constant changement, l'Internet des Objets (IoT) se distingue comme une avancée significative, modifiant radicalement notre vie de tous les jours grâce à diverses applications dans les secteurs de la santé, de l'industrie, de l'agriculture et même des cités intelligentes. Cependant, ce grand maillage d'objets crée de nouveaux enjeux, notamment en termes de sécurité, de protection des données et de capacité à résister aux menaces informatiques.

Cette recherche s'inscrit dans ce contexte en offrant une contribution majeure à la sécurisation des réseaux IoT grâce à l'intégration de l'intelligence artificielle. Suite à une analyse rigoureuse des failles spécifiques à l'univers de l'IoT et des méthodes typiques de détection, deux systèmes intelligents de détection d'intrusion ont été élaborés et mis en place :

Un système basé sur le Machine Learning (ML-NIDS), exploitant des algorithmes classiques (Random Forest, SVM, etc.) permettant une détection rapide et efficace des attaques connues et des anomalies.

Un système basé sur le Deep Learning (DL-NIDS), s’appuyant sur des architectures de réseaux neuronaux (MLP, CNN, RNN) capables d’identifier des schémas complexes et de détecter des attaques sophistiquées dans des environnements hétérogènes.

Tous ces modèles ont été soumis à des tests et évaluations sur le jeu de données CIC-IoT-2023, ce qui a permis de confirmer leur solidité et leur performance en matière de précision, de taux de détection et d'atténuation des faux positifs. Ces résultats

indiquent que les méthodes suggérées dépassent plusieurs modèles de référence actuels, et démontrent l'importance de l'intelligence artificielle dans la sécurité des réseaux IoT.

En résumé, ce travail de recherche apporte une contribution double : premièrement, il fournit une analyse approfondie et précise des problématiques de sécurité liées à l'IoT ; deuxièmement, il suggère des réponses tangibles, utiles et flexibles face aux défis émergents dans le domaine de la cybersécurité. Il ouvre aussi la voie pour des perspectives futures comme l'amélioration de modèles en temps réel, l'incorporation de la détection collaborative ou encore la résilience contre les attaques instantanées.

Bibliographie

[1] Information technology - Telecommunications and information exchange between systems — Near Field Communication — Interface and Protocol (NFCIP-1)", ISO/IEC 18092, First Edition, 2004-04-01.

[2] Ilie-Zudor, E., Kemény, Z., Egri, P., & Monostori, L. (2006). The RFID Technology and Its Current Applications. In Proceedings of the Modern Information Technology in the Innovation Processes of the Industrial Enterprises (MITIP 2006), 29–36. Budapest, Hungary: Computer and Automation Research Institute, Hungarian Academy of Sciences.

[3] Gosseaume, Audrey. (2025, 24 février). *Détecteur de présence : les avantages et inconvénients*. Le Figaro, MEMOIRE : Conception et réalisation d'un système domotique en utilisant des technologies IOT, l'Université Chouaib Doukkali (faculté des sciences ou ENSAJ)

[4] AltexSoft. *Internet of Things (IoT) Architecture: Key Layers and Components*. AltexSoft, Publication, MEMOIRE : Détection des risques et des menaces dans le domaine des IoT, Université de Tlemcen , Faculté de technologie

[5] Cheikh, I., Roy, S., Sabir, E., & Aouami, R. (2025).
Energy, Scalability, Data and Security in Massive IoT: Current Landscape and Future Directions

[6] http://theses.insa-lyon.fr/publication/2021LYSEI018/these.pdf © [J. tournier], [2021], INSA Lyon, tous droits réservés

[7] Phan, M. Q. (2012). Security in Wireless Sensor Networks: Preventing DoS Attacks in the RPL Protocol (Master's thesis). Edith Cowan University. https://ro.ecu.edu.au/theses/2303

[8] Sahai et Waters (Key Generation Server – KGS)

[9] Sfar, I., Natalizio, E., Challal, Y., & Chtourou, Z. (2018). A Roadmap for Security Challenges in the Internet of Things. Digital

Communications and Networks, 4(2), 118–137. https://doi.org/10.1016/j.dcan.2017.04.003

[10] https://www.spiceworks.com/tech/artificial-intelligence/articles/what-is-logistic-regression/

[11] https://xgboost.readthedocs.io/en/release_3.0.0/

[12] https://fr.mathworks.com/discovery/convolutional-neural-network.html

[13] Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey.ACM Computing Surveys (CSUR), 41(3), 1-58

[14] https://jupyter-notebook.readthedocs.io/en/stable/notebook.html

[15] https://www.cloudflare.com/fr-fr/learning/ddos/udp-flood-ddos-attack/

[16] https://www.cloudflare.com/learning/ddos/ping-icmp-flood-ddos-attack/

[17] Malwarebytes. (n.d.). Les attaques par force brute expliquées. Consulté le 4 juin 2025, à partir de https://www.malwarebytes.com/fr/cybersecurity/basics/brute-force-attack

[18] Cloudflare. (n.d.). Qu'est-ce que le botnet Mirai ?. Consulté le 4 juin 2025, à partir de https://www.cloudflare.com/fr-fr/learning/ddos/glossary/mirai-botnet/

[19] https://www.cloudflare.com/fr-fr/learning/ddos/glossary/denial-of-service/

[20] https://www.mailinblack.com/ressources/glossaire/quest-ce-que-le-spoofing/

[21] http://cisco.ofppt.info/ccna1/course/module11/11.2.2.2/11.2.2.2.html

[22] Qrator Labs. (2024). Statistiques et aperçu des attaques DDoS du premier trimestre 2024. Consulté le 4 juin 2025, à partir de https://blog.qrator.net/en/q1-2024-ddos-attacks-statistics-and-overview_198/

[23] https://www.unb.ca/cic/datasets/iotdataset-2023.html

[24] https://www.ibm.com/think/topics/random-forest

[25] https://www.geeksforgeeks.org/support-vector-machine-algorithm/

[26] https://developers.google.com/machine-learning/crash-course/logistic-regression/sigmoid-function?hl=fr

[27] Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 785–794). ACM.