



Université Abdelmalek Essaadi  
Faculté des Sciences et Techniques de Tanger  
Département Génie Informatique



# MEMOIRE DE PROJET DE FIN D'ETUDE MASTER SYSTEMES INFORMATIQUES & RESEAUX

Sujet

« APPLICATION DES RESEAUX DE NEURONES  
RECURRENTS SUR L'IOT POUR LA PREDICTION  
DES DONNEES METEOROLOGIQUES »

Réalisé par

Mlle. BEN ABDEL OUAHAB Ikram

Membres du Jury

Prof. EL AMRANI Chaker	Rapporteur
Prof. BEN ABDELOUAHAB Jaafar	Président
Prof. ASTITO Abdelali	Co-Encadrant
Prof. BOUHORMA Mohammed	Encadrant

Le 11 Juillet 2018

Année Universitaire 2017/2018

# RESUME

Ce projet adopte les réseaux de neurones récurrents avec son architecture Long Short-Term Memory (LSTM) pour prédire la température ambiante (TA). La prédiction se base sur les données météorologiques récupérées des stations IoT. Ces stations qui se composent de capteur de température, d'humidité ainsi que certains Gaz dans l'air. Les données captées sont envoyées à une station de base par le protocole Long Range (LoRa). Nous formulons le problème de prédiction de TA comme un problème de régression de séries chronologiques. LSTM est un type particulier de réseau de neurones récurrent, qui a une forte capacité à modéliser la relation temporelle des données de séries chronologiques et peut bien gérer le problème de dépendance à long terme. L'architecture de réseau proposée est composée de deux types de couches: couche LSTM et couche dense connectée complète. La couche LSTM est utilisée pour modéliser la relation de séries chronologiques. La couche entièrement connectée est utilisée pour mapper la sortie de la couche LSTM vers une prédiction finale. Pour confirmer l'efficacité du modèle proposée, nous réalisons des tests sur des données captées par notre propre système IoT sur Tanger. En outre, nous montrons tous les résultats dans une interface Web.

## MOTS CLES

*Internet des Objets, LoRa, apprentissage profond, Réseaux de neurones récurrents (RNN), LSTM.*

# ABSTRACT

This project adopts recurrent neural networks (RNN) with its Long Short-Term Memory (LSTM) architecture to predict the ambient temperature (TA). The prediction is based on meteorological data retrieved from IoT stations. These IoT stations consist of different components such as sensors to capture the temperature, humidity and some gases in the air, and send them to the basic station with LoRa protocol. We formulate the TA prediction problem as a time series regression problem. LSTM is a particular type of recurrent neural network, which has a strong ability to model the temporal relationship of time series data and can well manage the problem of long-term dependency. The proposed network architecture consists of two types of layers: LSTM layer and full connected dense layer. The LSTM layer is used to model the time series relationship. The fully connected layer is used to map the output of the LSTM layer to a final prediction. To confirm the effectiveness of the proposed model, we perform tests on data collected by our own IoT system on Tangier. In addition, we show all the results in a web interface.

## KEY WORDS

*Internet of Things, Long Range Wide Area Network (LoRaWAN), Deep Learning, Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM).*

# TABLE DES MATIERES

LISTE DES FIGURES	VI
LISTE DES TABLEAUX	VII
GLOSSAIRES DES ACRONYMES	VII
REMERCIEMENT	VIII
CHAPITRE I : INTRODUCTION GENERALE	9
1. CONTEXTE	10
2. INTERNET DES OBJETS	10
3. DEEP LEARNING	11
4. PROBLEMATIQUE	11
5. OBJECTIF	12
6. ORGANISATION DU MANUSCRIT	12
CHAPITRE II : ETAT DE L'ART	13
1. L'INTERNET DES OBJETS	14
1.1. Introduction	14
1.2. Historique de l'Internet des objets	14
1.3. Qu'est-ce qu'un objet connecté ?	15
1.4. La communication entre les objets connectés	16
1.4.1. Les technologies de courte portées	16
a. Le protocole NFC	16
b. Bluetooth	16
c. Zigbee	16
1.4.2. Les technologies de moyenne portée	17
a. Z-Wave	17
b. Wi-Fi	17
c. Bluetooth Low Energy	17
1.4.3. Les technologies de longue portée	18
a. Réseaux cellulaires mobiles	18
b. Réseaux bas-débit « LPWAN » : Sigfox & LoRa	18
1.5. Avantages & Enjeux de l'IoT	19
1.6. Application de l'Internet des Objets	20
1.7. Exemples réels de déploiement	21
1.7.1. Parking Intelligent	21
1.7.2. Smart Grid & metering	21
2. RÉSEAUX LOW POWER WIDE AREA NETWORK	22
2.1. Définition	22
2.2. Caractéristiques	23
2.3. Technologies	23
2.4. Architecture	23
3. RÉSEAUX LONG RANG WIDE AREA NETWORK	24
3.1. Définition	24
3.2. Architecture du réseau	24
3.3. Concept de sécurité	25
3.4. Exemples d'utilisation réels	26
3.4.1. Smart City	26
3.4.2. Monitoring industriel	26
3.4.3. Agriculture	26
4. L'ANALYSE DE DONNEES	27
4.1. Introduction	27
4.2. Apprentissage profond -Deep learning-	28
4.3. Qu'est-ce qu'un réseau de neurones ?	29
4.4. Architectures de l'apprentissage profond	30
4.4.1. Réseaux de Neurones Profonds (DNN)	30
4.4.2. Réseaux de neurones récurrents (RNN)	30

4.4.3.	Long short-Term Memory (LSTM)	31
4.4.4.	Gated Recurrent Unit (GRU)	31
a.	Description des étapes du fonctionnement des RNN & LSTM	31
<b>CHHAPITRE III : METHODOLOGIE</b>		<b>33</b>
1.	OUTILS MATERIELS	34
1.1.	Raspberry Pi 3	34
1.2.	Capteur de Température et d'Humidité	35
1.3.	Capteur de Gaz	36
1.4.	Wifi Lora 32	36
1.4.1.	Spécifications techniques	36
1.4.2.	Tests de distance	37
1.4.3.	Démonstration	38
a.	Test 1	38
b.	Test 2	39
c.	Test 3	39
d.	Résultat Final	40
2.	FRAMEWORKS DE L'APPRENTISSAGE PROFOND	40
3.	OUTILS LOGICIELS	41
3.1.	Arduino IDE	41
3.2.	Bibliothèques Python	42
<b>CHAPITRE IV : IMPLEMENTATION</b>		<b>43</b>
1.	LORA & CAPTEURS	44
1.1.	Le prototype réalisé	45
1.2.	Fonctionnement du système	46
1.3.	Fonctionnalités de la Raspberry	46
2.	APPRENTISSAGE PROFOND ET PREDICTIONS	47
2.1.	Prédiction de la Température Simple	47
2.1.1.	Problématique	47
2.1.2.	Architecture du modèle	47
2.1.3.	Description du modèle	48
2.1.4.	Résultat	48
2.2.	Prédiction de la pollution dans l'air	49
2.2.1.	Description des données	49
2.2.2.	Architecture du modèle	49
2.2.3.	Application avec Keras	50
2.2.4.	Résultat	51
2.3.	Prédiction de la Température avancée	51
2.3.1.	Description des données	51
2.3.2.	Architecture du modèle	52
2.3.3.	Application avec KERAS	52
a.	Tâches réaliser sur un PC	52
b.	Résultats d'entraînement	53
c.	Tâches Réaliser sur La Raspberry Pi 3	53
3.	VISUALISER LES DONNEES	54
<b>CHAPITRE V : RESULTATS &amp; DISCUSSION</b>		<b>55</b>
1.	RESULTAT DE LA PREDICTION	56
1.1.	Comparaisons	57
1.1.1.	Variation de couches cachées	57
1.1.2.	Taille de la base de données	58
2.	INTERFACE WEB DE LA RASPBERRY PI	59
<b>CONCLUSION GENERALE</b>		<b>62</b>
<b>REFÉRENCES</b>		<b>63</b>

# LISTE DES FIGURES

FIGURE 1 - LES OBJETS CONNECTES A LA CONQUETE DE NOTRE QUOTIDIEN	15
FIGURE 2 - TECHNOLOGIES DE COURTE PORTEE	16
FIGURE 3 - TECHNOLOGIES DE MOYENNE PORTEE	17
FIGURE 4 - TECHNOLOGIES DE LONGUE PORTEE	18
FIGURE 5 - DIFFERENTS PROTOCOLES DE COMMUNICATION ENTRE LES OBJETS CONNECTES	18
FIGURE 6 - COMPTEUR ELECTRIQUE INTELLIGENT, PAR ERFD	21
FIGURE 7 - COMPARAISON DE LPWAN, RESEAU CELLULAIRES MOBILES ET ZIGBEE	22
FIGURE 8 - EXEMPLE D'ARCHITECTURE D'UN RESEAU LPWAN	23
FIGURE 9 - ARCHITECTURE D'UN RESEAU LoRAWAN	24
FIGURE 10 - LE PROCESSUS DE SECURITE DANS LES RESEAUX LoRAWAN	25
FIGURE 11 - ANALYSE DE DONNEES COLLECTES DE L'IOT	27
FIGURE 12 - LES BRANCHES DE L'INTELLIGENCE ARTIFICIELLE	28
FIGURE 13 - LA MISE EN CORRESPONDANCE NEURONE BIOLOGIQUE / NEURONE ARTIFICIEL	29
FIGURE 14 - ARCHITECTURE GENERALE D'UN RESEAU DE NEURONES	30
FIGURE 15 - ARCHITECTURES DE L'APPRENTISSAGE PROFOND DE 1990 JUSQU'A 2015	30
FIGURE 16 - FONCTIONNEMENT DES RNN & LSTM	31
FIGURE 17 - LES OUTILS MATERIELS	34
FIGURE 18 - RASPBERRY PI 3 MODEL B	34
FIGURE 19 - RASPBERRY PI 3 MODEL B+ (VERSION 2018)	34
FIGURE 20 - CAPTEUR DE TEMPERATURE ET D'HUMIDITE DHT11	35
FIGURE 21 - CAPTEUR DE GAZ MQ2	36
FIGURE 22 - MODELE WIFI LoRa 32	36
FIGURE 23 - WIFI LoRa 32 PINOUT DIAGRAM	37
FIGURE 24 - FORME DU PAQUET LoRa	37
FIGURE 25 - INSTALLATION DE WIFI LoRa 32 DANS ARDUINO IDE	41
FIGURE 26 - SCHEMA GENERALE DU SYSTEME IOT	44
FIGURE 27 - MONTAGE ELECTRIQUE DES CAPTEURS AVEC LE MODULE LoRa	44
FIGURE 28 - LE PROTOTYPE DE LA STATION IOT	45
FIGURE 29 - PROTOTYPE, STATION B	45
FIGURE 30 - PROTOTYPE, STATION A	45
FIGURE 31 - ARCHITECTURE DU MODELE DE LA PREDICTION DE TEMPERATURE SIMPLE	47
FIGURE 32 - RESULTAT DE LA PREDICTION DE LA TEMPERATURE SIMPLE	48
FIGURE 33 - ARCHITECTURE DU MODELE DE LA PREVISION DE LA POLLUTION DANS L'AIR	49
FIGURE 34 - LA VARIATION DE L'ECART D'ERREUR DURANT L'APPRENTISSAGE DU MODELE	51
FIGURE 35 - ARCHITECTURE DU MODELE DE PREDICTION DE LA TEMPERATURE AVANCEE	52
FIGURE 36 - LA VARIATION DE L'ECART D'ERREUR	53
FIGURE 37 - RESULTAT D'ENTRAINEMENT DU MODELE, VALEUR DU RMSE	53
FIGURE 38 - LA TEMPERATURE EN ENTREE ET LES PREDICTIONS	56
FIGURE 39 - COMPARAISON DES PREDICTIONS AVEC LES VALEURS ORIGINALS	56
FIGURE 40 - RMSE PAR EVOLUTION DE COUCHE CACHEE	57
FIGURE 41 - RESULTAT DE LA PREDICTION DE POLLUTION	58
FIGURE 42 - RESULTAT DE LA PREDICTION DE TEMPERATURE	58
FIGURE 43 - COMPARAISON DE LA TAILLE DE LA BASE DE DONNEES ET LE NOMBRE D'EPOCHS	58
FIGURE 44 - INTERFACE WEB, PRESENTATION	59
FIGURE 45 - INTERFACE WEB, STRUCTURE	59
FIGURE 46 - INTERFACE WEB, VISUALISATION DES DONNEES (1)	60
FIGURE 47 - INTERFACE WEB, VISUALISATION DES DONNEES (2)	60
FIGURE 48 - INTERFACE WEB, PREDICTION (1)	61
FIGURE 49 - INTERFACE WEB, PREDICTION (2)	61
FIGURE 50 - INTERFACE WEB, PREDICTION (3)	61

# LISTE DES TABLEAUX

TABLEAU 1 - DOMAINES D'APPLICATION DE L'IIOT	20
TABLEAU 2 - CARACTERISTIQUES DES RESEAUX LPWAN	23
TABLEAU 3 - COMPARAISON DES ARCHITECTURES LSTM	57

# GLOSSAIRES DES ACRONYMES

AES	Advanced Encryption Standard
AI	Intelligence Artificielle
DL	Deep Learning / Apprentissage profond
DNN	Deep Neural Network
GRU	Gated Recurrent Unit
IoT	Internet of Things / Internet des objets
LoRa	Long Range
LoRaWAN	Long Range Wide Area Network
LPWAN	Low Power Wide Area Network
LSTM	Long Short-Term Memory
ML	Machine Learning / Apprentissage automatique
NN	Neural Network
RNN	Recurrent Neural Network / Réseau de neurone récurrent
RPi	Raspberry Pi
TA	Température Ambiante

# REMERCIEMENT

Le travail présenté dans ce manuscrit est le résultat de quatre mois de travail réalisé dans le cadre d'un projet de fin d'étude pour l'obtention du Master Systèmes Informatiques et Réseaux. Ce projet a été effectué au sein de la Faculté des sciences et techniques de Tanger, au département Informatique sous l'encadrement du Prof. *MOHAMMED BOUHORMA*.

Je remercie chaleureusement mon encadrant, Prof. *Mohammed BOUHORMA*, professeur au département Informatique, qui m'a fait confiance tout au long de ma formation et spécifiquement ces mois de stage. Il a su créer des conditions de travail idéales malgré sa présence réduite dans les locaux de la FSTT, alliant une très grande liberté et la participation à plusieurs événements scientifiques. Pour tout cela, je le remercie vivement.

Je tiens à remercier également mon co-encadrant Prof. *Abdelali ASTITO*, professeur et chef du département Génie Electrique, pour les discussions techniques et moins techniques, et également aux matériels qui nous a mis à disposition pour la réalisation de ce projet.

Toutefois je suis honorée par l'intérêt porté à ce rapport par les membres du jury qui ont pris la peine d'évaluer cette étude, auxquels je présente ma gratitude et mon profond respect.

Mes derniers remerciements s'adressent à mes parents, et toute ma famille qui m'ont toujours soutenu d'une manière exemplaire, ainsi que mes collègues.



# CHAPITRE *I*

## INTRODUCTION GENERALE



# 1. CONTEXTE

Plus que la moitié de la population ont choisi de vivre dans les villes, un chiffre qui devrait atteindre 70% d'ici 2050 (d'après L'Organisation des Nations Unies). Le succès a toujours un coût, et celui des villes ne fait pas l'exception. Les modes de consommation non viables, la dégradation de l'environnement et les inégalités persistantes ne sont que quelques-uns des problèmes qui affligent les villes modernes.

Alors que les villes occupent, aujourd'hui, 2 % de la surface du globe, elles abritent 50 % de la population mondiale, consomment 75 % de l'énergie produite et sont à l'origine de 80 % des émissions de CO<sub>2</sub>. À la fois énergivore et principale source des émissions de CO<sub>2</sub>, la ville et ses habitants sont les premiers concernés par les dangers du **réchauffement climatique**.

C'est pour ces raisons que les villes sont considérées comme des hauts lieux de la bataille contre le changement climatique. L'Union européenne a estimé à 11 milliards d'euros les investissements publics et privés nécessaires pour permettre à 25 millions d'Européens de vivre dans des **villes intelligentes**. Au Maroc également, **Casablanca** est depuis octobre 2015, la première ville africaine à faire partie du réseau de 25 villes intelligentes sélectionnées IEEE (Institute of Electrical and Electronic Engineers), la plus importante association de professionnels du digital et des technologies de l'information au monde. Cette organisation appuiera Casablanca dans le développement de son concept de ville intelligente sociale et frugale.

# 2. INTERNET DES OBJETS

On assiste récemment à l'émergence de l'**Internet des objets**, qui est un nouveau paradigme bouleversant le domaine des réseaux de télécommunication. L'Internet des objets qui est une partie intégrante dans l'Internet du futur, consiste en une large interconnexion de toutes sortes d'objets (autre que les ordinateurs et les téléphones mobiles) dans notre entourage, à titre d'exemple les véhicules, les routes, la maison, la télévision, les vêtements, et bien d'autres pour un monde ambiant et intelligent.

La différence essentielle entre "Internet" et "Internet des Objets" est que dans ce dernier, il y a juste « **moins de tout** » dans un périphérique donné : moins de mémoire, moins de puissance de traitement, moins de bande passante et aussi moins d'énergie disponible. C'est soit parce que les "objets" disposent d'une batterie et que maximiser la durée de vie est une priorité ou parce que le nombre des objets est énorme (*on estime qu'il y aura 50 milliards d'appareils connectés d'ici 2020*). Cette volonté de « faire plus avec moins » conduit à des contraintes qui limitent l'applicabilité des réseaux cellulaires traditionnels, ainsi que des technologies, telles que Wi-Fi, en raison des exigences d'énergie et d'évolutivité.

Une autre gamme de protocoles et de technologies est apparue pour répondre aux exigences de communication de l'IoT, comme **les réseaux étendus de faible puissance (LPWAN)** mentionné dans [1], et les **LTE based Cellular IoT Network** apparue dans [2]. Familièrement parlant, un LPWAN est censé être à l'IoT ce que le WiFi était au réseautage grand public.

### 3. DEEP LEARNING

Le **Deep Learning** suscite un intérêt majeur, après des résultats spectaculaires, dont la victoire d'AlphaGo sur le meilleur joueur mondial de Go. Inspiré au départ par une métaphore biologique, celle du cerveau, le domaine des réseaux de neurones est devenu l'un des principaux axes de recherche de l'intelligence artificielle (IA). On arrive, grâce à des algorithmes inspirés par les neurones du cerveau, à réaliser des programmes capables d'apprendre à partir d'exemples et de généraliser ensuite ce qu'ils ont appris à reconnaître.

Les applications des réseaux de neurones ont vu le jour depuis quelques années dans plusieurs domaines : d'abord, dans l'identification d'images, et la classification. Ensuite, dans la conversion/traduction séquence à séquence, ou comment convertir la parole en un texte et comment passer d'une langue à une autre. Les premiers sont réalisés la plupart du temps avec des réseaux de neurones convolutionnels, et les seconds sont réalisés avec des réseaux de neurones récurrents, dont en particulier les **Long Short-Term Memory (LSTM)** et les **Stacked LSTM**.

Nous nous intéressons tout particulièrement aux **RNN** avec leurs architectures profondes **Stacked LSTM** qui permettent de s'affranchir d'un certain nombre de difficultés rencontrées avec des RNN standards.

### 4. PROBLEMATIQUE

Dans ce contexte, rendre les villes intelligentes et durables, c'est essayer de diminuer l'impact environnemental, mais, également, de repenser en profondeur les modèles d'accès aux ressources, les transports, la gestion des déchets, la climatisation des édifices et surtout la gestion de l'énergie.

Autrement, le terme d'« **Intelligence artificielle** » et l'idée de construire des réseaux de neurones artificiels ne sont pas nouveaux : ils remontent à la fin des années 40-50. À cette époque, les notions d'intelligence artificielle et d'apprentissage n'étaient pas encore corrélées : les programmes étaient conçus pour prendre en compte un certain nombre de scénarios, prédéfinis par les développeurs. Cette approche a donc rapidement fait apparaître ses limites.

Il a fallu attendre les années 1990 avec l'apparition de nouvelles technologies de **Machine Learning**, puis 2000 avec la démocratisation de moyens de calculs de plus en plus puissants, et 2010 avec l'accès au **Big Data** et l'accessibilité de **GPU** extrêmement puissantes, pour avoir la possibilité d'atteindre des couches de plus en plus denses et profondes des réseaux de neurones. C'est depuis 2012 que se mettent en place de gros projets d'intelligence artificielle, portés par IBM (Watson), Google (DeepMind/AlphaGo) et Facebook (DeepFace).

*Aujourd'hui comment on peut appliquer l'Intelligence artificielle aux villes intelligentes ? Comment préparer les villes intelligentes aux différents défis environnementaux pour garantir aux générations futurs un cadre de vie aussi bon que celui actuel ? Est-il possible de mettre en place un système d'objets connectés au sein des villes intelligentes capable de capter des données, des habitudes et des comportements ? Quels seront les bénéfices de la prédiction de ces comportements, et différents données des villes intelligentes ? Et est-ce que ces prédictions peuvent aider à prévenir les habitants en cas d'incidents naturels ou autrement à la détection des anomalies ?*

## 5. OBJECTIF

L'objectif principal de cette étude est l'application des réseaux de neurones profond sur des tâches spécifiques de prédiction de différentes données météorologiques basé sur les collections des capteurs dans les villes intelligentes. Ainsi que la proposition de nouvelles solutions à deux niveaux : d'abord pour la collecte des données dans le cadre de l'Internet des Objets, ensuite pour améliorer les performances de méthode d'apprentissage profond, l'adapter et l'intégrer dans le système IoT réalisé.

Premièrement, la réalisation d'un prototype basé sur les capteurs et différents composants qui permettent l'implémentation de l'Internet des Objets dans les villes intelligentes, dont le but principale est la collecte des données météorologiques. Les composants du système communiquent via un réseau de longue portée et basse consommation, qui garantit le transfert périodique des données de plusieurs lieux.

Ensuite, dans le cadre de l'analyse prédictive des données collectées, nous introduisons une modification de différents modèles d'apprentissage profond avec divers architectures, tout en les adaptant avec nos propres données. Et, pour améliorer leur comportement, il est nécessaire de faire varier l'ensemble des paramètres qui les influence.

Notre but est de minimiser l'écart d'erreur (RMSE) dans les deux phases ; d'entraînement et de test, afin d'avoir un bon modèle. Ce modèle résultant, intitulé Stacked Long Short-Term Memory (Stacked LSTM) est comparé au modèle LSTM standard ainsi qu'un réseau de neurones basic.

Finalement, pour la visualisation et l'accès au système IoT ainsi qu'aux prédictions réalisé par le modèle d'apprentissage profond, le prototype est accessible via une interface Web qui facilite et simplifie la visualisation de tous ces résultats.

## 6. ORGANISATION DU MANUSCRIT

Ce manuscrit est organisé en cinq chapitres.

Le premier chapitre introduit le contexte, la problématique et les objectifs de l'étude.

Le Chapitre II présente le cadre théorique du sujet, au travers d'une présentation générale de l'IoT ainsi que les protocoles de communication entre les objets connectés et spécifiquement LoRa sur lequel nous avons travaillé. Et la dernière partie de ce chapitre présente les différentes technologies d'analyse de données, et particulièrement les architectures et les algorithmes du Deep Learning.

Le Chapitre III est consacré à la présentation des outils matériels et logiciels, qui ont servi à la réalisation du projet.

Le Chapitre IV détaille le processus de développement des deux parties du projet, la première dédié à l'IoT et décrit le fonctionnement des stations IoT. Et la deuxième, décrit le modèle d'apprentissage profond permettant la prédiction de la température, d'après les données météorologiques collectées par les stations IoT.

Finalement, le Chapitre V présente les résultats de prédiction dans l'interface Web du système IoT.

## CHAPITRE *II*

### ETAT DE L'ART



# 1. L'INTERNET DES OBJETS

## 1.1. INTRODUCTION

L'**Internet des objets** ou **Internet of Things (IoT)** est un paradigme émergeant dans le monde des réseaux informatiques. Il existe plusieurs définitions de l'**IoT** dans la littérature.

Il peut être défini comme une évolution et extension de l'Internet de nos jours pour l'inclusion de tous les objets et les endroits dans notre entourage (réfrigérateurs, thermostat, maisons, véhicules, routes, etc.). Le concept prometteur de l'IoT va simplifier notre vie, nous faire gagner du temps, décharger notre cerveau de la mémorisation de données logistiques (itinéraires, temps de prise des médicaments, etc.). Ainsi, l'accès ubiquitaire à différents types d'informations permettrait la sophistication du mode de vie et l'amélioration significative de la qualité des services dans différents domaines.

L'**IoT** est également défini comme un paradigme de communication récent qui envisage un futur proche. Dans lequel les objets de la vie quotidienne seront équipés de microcontrôleurs, d'émetteurs-récepteurs pour la communication numérique et de piles de protocoles appropriés pour communiquer entre eux et avec les utilisateurs [3].

Comme l'**IoT** vise à interconnecter les gens, les données et tous les objets, on peut dire qu'il y ait une fusion entre le monde réel (physique) et le monde numérique (virtuel). Les objets du monde physique vont être incorporés dans le monde virtuel de l'Internet. Cela fait appel à de nouvelles tendances et innovations que ce soit sur le plan architectures de communications ou sur le plan présentation et exploitation des services.

## 1.2. HISTORIQUE DE L'INTERNET DES OBJETS

L'émergence de l'Internet des objets n'est que le résultat de convergence entre multiples technologies, à savoir l'Internet, la communication sans fil, les systèmes embarqués, systèmes micro-électroniques et la nanotechnologie. Dans cette section, nous citons les événements les plus marquants sur le chemin de la concrétisation de l'IoT.

Le concept d'un réseau de dispositifs intelligents a été évoqué pour la première fois en 1982, avec le premier appareil connecté à Internet à l'Université Carnegie Melon capable de signaler à son inventaire si les boissons nouvellement chargées sont bien froides [4]. Ainsi, en 1991, Mark Weiser a introduit l'informatique omniprésente à travers son papier [5] intitulé : « **The computer for the 21<sup>st</sup> Century** » et a présenté d'avance la vision contemporaine de l'Internet des objets. Un peu plus tard, en 1994, Steve Mann avait créé le *WearCam* qui était parmi les premières caméras à apparaître sur le web [6]. Ensuite, en 1998, l'informatique ubiquitaire a commencé d'attirer l'attention par le fait qu'elle permettrait l'incorporation flexible et efficace de l'informatique dans la vie quotidienne.

En 1999, la désignation Internet des objets a été prononcée pour la toute première fois par Kevin Ashton [7]. Après, en 2000 la société LG annonce son premier réfrigérateur intelligent connecté à Internet. De plus, la technologie **RFID** (*Radio Frequency IDentification*) qui est l'une des technologies constitutionnelles de l'IoT, a commencé à être massivement déployée vers les années 2003 et 2004. D'autre part, une initiative très intéressante a été prise en 2008 ; un groupe de recherche appelé IPSO Alliance [8] s'est consacré à promouvoir l'utilisation du protocole IP (*Internet Protocol*) pour les réseaux d'objets miniatures intelligents.

De nombreux travaux de recherches ont été succédés et se sont tous concentrés autour de la réalisation, dans les meilleures conditions, de la vision de l'Internet des objets et la mener à sa maturité en dépit de tous les défis soulevés. Cela avec la considération des progrès technologiques continus dans le marché des dispositifs intelligents et dans le domaine de technologies de télécommunication comme : le *Cloud Computing*, le concept du SDN (*Software-Defined Networking*), etc.

### 1.3. QU'EST-CE QU'UN OBJET CONNECTÉ ?

Avec l'avènement de l'IoT, la connexion Internet acquiert une troisième dimension; en plus de la possibilité de se connecter n'importe quand et n'importe où, il est désormais possible d'être connecté avec n'importe quel objet. De plus, les objets connectés sont identifiés de façon unique et sont capable de récolter des informations environnementales (*liées aux changements des paramètres de l'environnement*) ou comportementales (*issues des variations d'état de l'objet lui-même ou des objets contextuels*), de les traiter et de les communiquer sur Internet. D'où vient leur appellation par **objets intelligents** ou encore **objets connectés**.

Cisco prévoit que d'ici quelques années, spécifiquement en 2020, l'Internet des objets sera une réalité et le nombre d'objets connectés dépassera les 50 milliards. A ce stade, il est nécessaire de noter que les données massives générées par un nombre immense d'objets intelligents connectés présentent une source de la charge globale de données qualifiées de *Big Data* sur Internet.

L'Internet des Objets fait référence à une gamme croissante de dispositifs connectés qui envoient des données sur Internet. Un "objet" intègre de l'électronique capable de transférer des données sur un réseau, sans aucune interaction humaine [9]. On peut citer par exemple les **wearables**, les **capteurs environnementaux**, les **machines d'usine**, les **appareils** installés dans les habitations et les bâtiments ou encore **les composants d'un véhicule**.

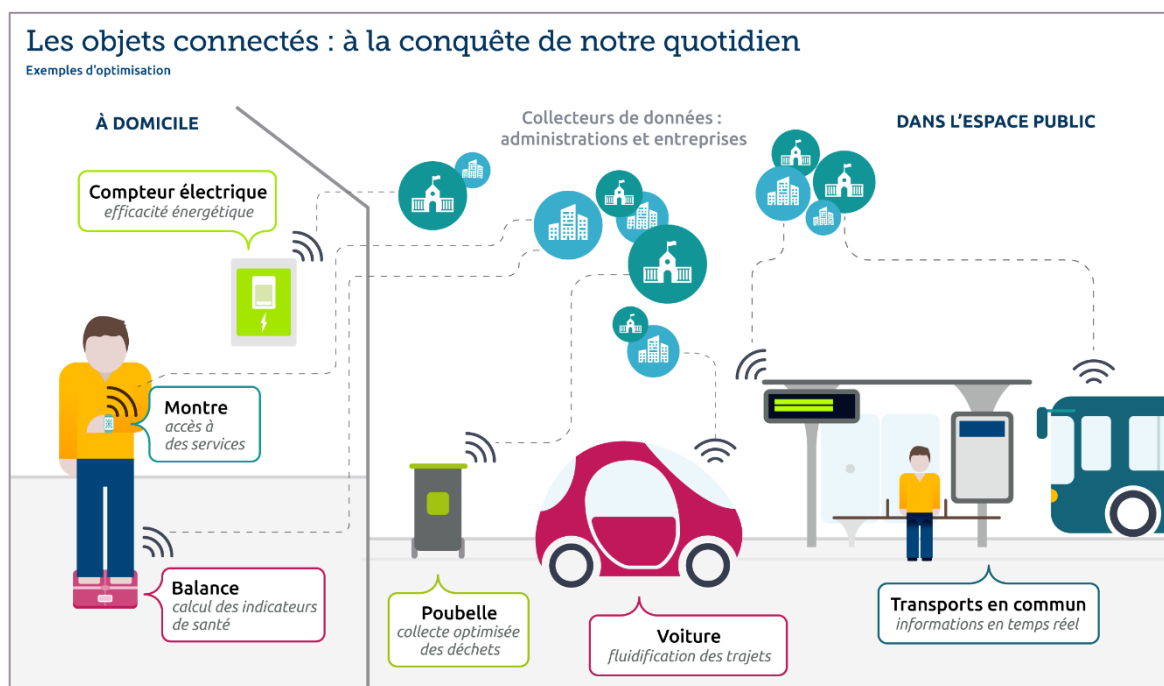


Figure 1 - Les objets connectés à la conquête de notre quotidien

## 1.4. LA COMMUNICATION ENTRE LES OBJETS CONNECTES

Pour communiquer, les objets ont besoin d'un réseau. Le choix est vaste et continue à grandir pour répondre aux différentes exigences des objets. Pour l'instant, les réseaux sont segmentés par portée, de la façon suivante : courte, moyenne et longue portée.

### 1.4.1. LES TECHNOLOGIES DE COURTE PORTEES

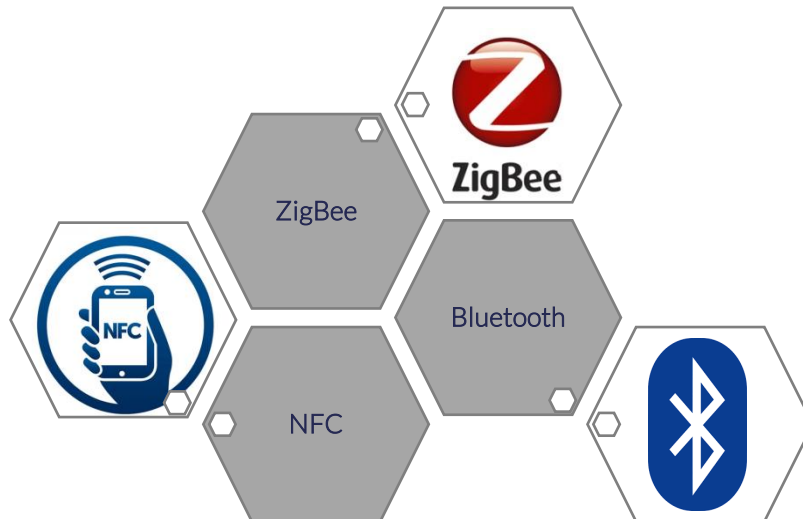


Figure 2 - Technologies de courte portée

#### a. LE PROTOCOLE NFC

Les protocoles **Near Field Communication (NFC)** sont fondés sur la technologie d'identification par radio fréquence RFID (Radio frequency identification). Le protocole NFC est un standard de communication radiofréquence sans contact à très courte distance, de l'ordre de quelques centimètres, permettant une communication simple entre deux équipements électroniques. Il est par exemple utilisé dans de nombreuses entreprises pour les badges d'accès aux locaux, ou comme support d'un abonnement à un réseau de transport en commun.

#### b. BLUETOOTH

Le protocole **Bluetooth** est un standard de transfert de données sans fil. Il utilise une faible bande passante, ce qui ne lui permet de transférer que peu de données à de courtes distances, mais est également très peu énergivore. Inclus à l'immense majorité des téléphones mobiles, afin de réaliser une communication entre deux téléphones, ou entre un téléphone et un objet connecté de nature différente. Il possède désormais de nombreuses applications : oreillette de discussion téléphonique sans fil, montre intelligente, moniteur de fréquence cardiaque, enceinte portative de diffusion de musique, station météo, thermostat, etc.

#### c. ZIGBEE

**Zigbee** est un protocole de communication radio développé spécifiquement pour les applications de domotique. D'une portée moyenne de 10 mètres, il utilise une faible bande passante et est idéal pour le transfert de données en faible volume. Peu énergivore et conçu pour des échanges de données à bas débit, le dispositif Zigbee convient aux appareils alimentés par une pile ou une batterie, et en particulier aux capteurs.



### 1.4.2. LES TECHNOLOGIES DE MOYENNE PORTEE

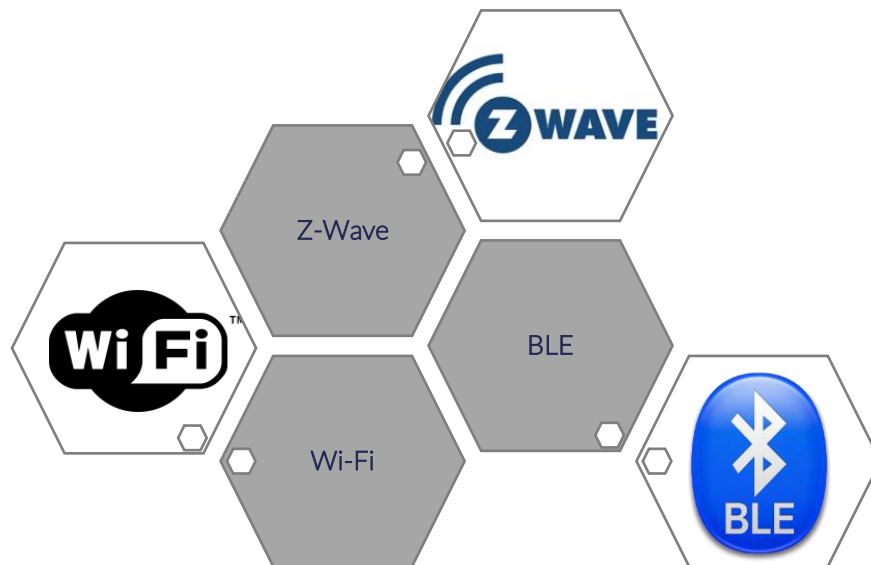


Figure 3 - Technologies de moyenne portée

#### a. Z-WAVE

Le **Z-Wave** est un protocole de communication sans fil principalement dédié à la domotique. Il permet de transmettre des données sur des distances allant de 30 mètres en intérieur à 100 mètres en plein air. Il fonctionne en réseau maillé, chaque appareil connecté pouvant relayer les informations émises par ses voisins, ce qui lui permet d'élargir sa portée. Tout comme le protocole Zigbee, l'utilisation de Z-Wave ne nécessite que très peu de puissance et les appareils peuvent donc communiquer pendant plusieurs années avec une simple pile.

#### b. WI-FI

Le **Wi-Fi** désigne un ensemble de protocoles de communications sans fil, permettant des connexions à **haut débit** sur des distances de 20 à 100 mètres. Il s'agit d'un réseau local sans fil très énergivore, qui ne convient que pour les appareils branchés sur secteur ou dont l'alimentation électrique peut être aisée et fréquente. Il permet de transférer rapidement beaucoup de données. Il existe différentes normes Wi-Fi correspondant à une portée et un débit variables.

#### c. BLUETOOTH LOW ENERGY

La technologie **Bluetooth Low Energy** (BLE) est un protocole de réseau personnel sans fil à très basse consommation. Comme la technologie Bluetooth originelle, le BLE ne permet de transférer qu'une quantité limitée de donnée à une distance moyenne de 60 mètres. La différence entre les dispositifs Bluetooth et BLE se situe au niveau de la consommation électrique nécessaire à la communication, qui est dix fois moindre pour BLE.

### 1.4.3. LES TECHNOLOGIES DE LONGUE PORTEE

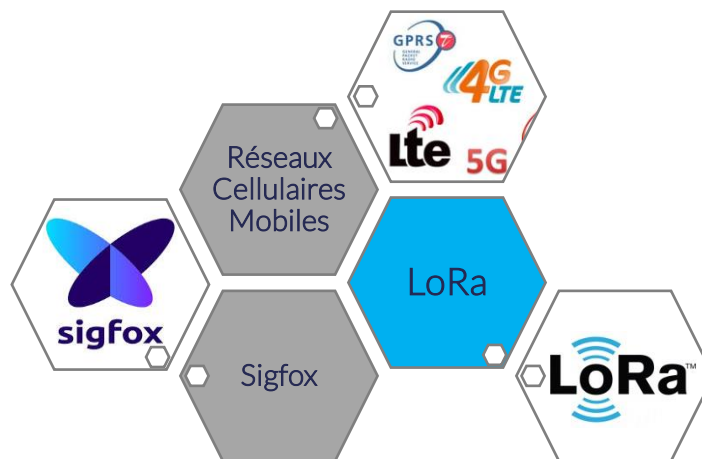


Figure 4 - Technologies de longue portée

#### a. RESEAUX CELLULAIRES MOBILES

Fournis par les opérateurs de télécommunication, les **réseaux cellulaires mobiles**, basés sur la technologie GSM, permettent de transférer une quantité importante de données à une longue portée. Ils nécessitent l'installation d'une carte SIM dans l'appareil à connecter, afin d'identifier celui-ci sur le réseau de communication.

#### b. RESEAUX BAS-DEBIT « LPWAN » : SIGFOX & LORA

**Sigfox** est un réseau de communication radio sans fil à bas débit et à basse fréquence, d'une portée moyenne de 10 kilomètres en milieu urbain et de 30 à 50 kilomètres en milieu rural. Ce réseau convient à des appareils à basse consommation, dotés ainsi d'une grande autonomie, qui transfèrent une faible quantité de données [10].

**LoRa** est un protocole de communication radio à très basse consommation, qui permet de transmettre des données en petite quantité, à des distances de 2 à 5 kilomètres en ville et jusqu'à 45 kilomètres en milieu urbain. À l'instar de Sigfox, il s'agit d'un dispositif qui convient particulièrement aux équipements peu énergivores n'émettant que périodiquement, notamment les capteurs.

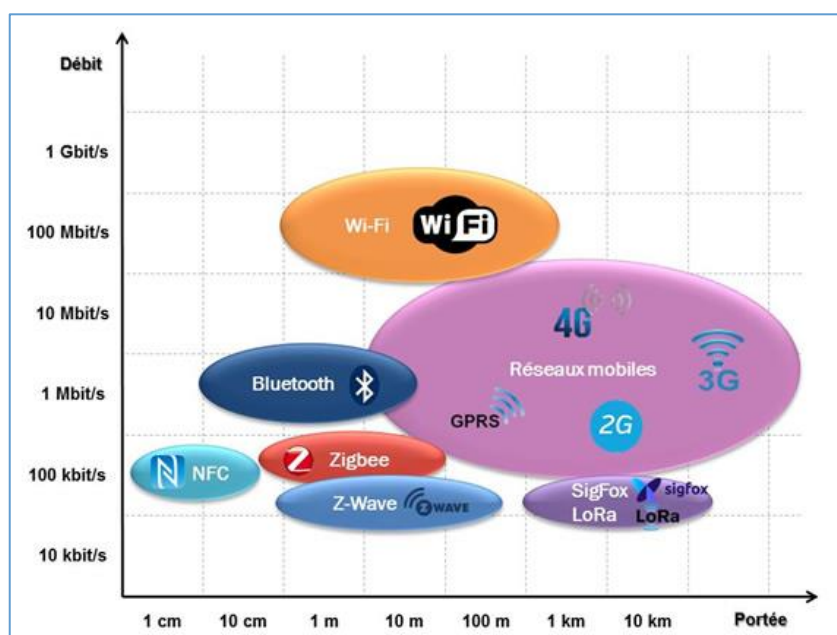


Figure 5 - Différents protocoles de communication entre les objets connectés

## 1.5. AVANTAGES & ENJEUX DE L'IOT

---

Les principaux avantages de l'IoT sont :

- Accès ubiquitaire à l'information pour un monde plus intelligent et un mode vie sophistiqué et confortable.
- Amélioration de la qualité de service et de la télésurveillance dans différents domaines d'applications.
- Améliorer la productivité et l'expérience-client : les objets connectés envoient des rapports à leurs constructeurs indiquant les préférences et les habitudes des clients aidant davantage les entreprises à agir de manière proactive et adaptée, qui satisfait la demande et les exigences de la clientèle.
- Le gain du temps est un autre avantage de l'IoT. Les déplacements inutiles sont dès lors remplacés par une simple navigation sur le web pour commander des produits, contrôler l'état des objets et/ou endroits connectés.
- Rationaliser nos dépenses et faire des économies vu qu'on ne consomme qu'en cas de besoin, que ça soit pour les achats ou la consommation énergétique (nécessaire pour l'éclairage ou la climatisation) ou autre.
- Possibilité d'exploitation des ressources géantes de l'Internet pour le stockage et le traitement des données écoulées de l'IoT.

Bien que l'Internet des objets soit un concept qui est à la fois avantageux et prometteur, et qui pourra apporter des solutions efficaces des problèmes du suivi et de télésurveillance dans différents domaines. En contrepartie, l'IoT soulève certaines questions décisives, étroitement liées à sa maturité et son acceptabilité.

On cite ci-dessous les enjeux les plus marquants :

- Sécurité
- Protection de la vie privée des utilisateurs
- Limitations de ressources
- Hétérogénéité
- Interopérabilité
- Transparence
- Le nombre croissant d'objets connectés
- Virtualisation
- Mobilité
- Qualité de service au niveau de la communication entre les objets

## 1.6. APPLICATION DE L'INTERNET DES OBJETS

L'Internet des objets n'est qu'un immense ensemble d'objets intelligents interconnectés et connectés à Internet. L'existence des objets intelligents avec de nouvelles possibilités de communications automatiques et intelligentes vont sensiblement améliorer le mode de vie des gens ainsi que la qualité de services dans divers domaines, à travers des degrés élevés d'autonomie et d'intelligence.

Citons quelques exemples de ses domaines d'application :

Domotique	<p>L'une des applications phares, qui touche un large public, est certainement la maison intelligente ou « bâtiment intelligent », avec ses nombreux objets communicants qui vont offrir de nouveaux services domotique, pour un meilleur contrôle des équipements et une utilisation optimale de l'énergie.</p> <p><i>Par exemple, la température ambiante intérieure sera contrôlée à la fois en fonction de la température extérieure et en fonction de la présence de personnes dans la maison. De même, éclairer, climatiser des locaux en l'absence de personne physique dans la maison sera inutile.</i></p>
Transport	<p>De nos jours, les voitures embarquant de nombreux équipements électroniques seront bientôt communicantes, et pourront dialoguer entre elles, ainsi qu'avec les infrastructures de la route. Cet échange permettra d'améliorer la sécurité routière en offrant des informations pertinentes en temps réel aux conducteurs, pour éviter des accidents ou des embouteillages. Cette meilleure gestion des transports aura un impact très fort sur l'économie.</p>
Industrie	<p>Le monde du travail et le monde industriel seront aussi impactés par les objets communicants connectés. Les usines intelligentes, truffées de capteurs communicants, collecteront des données en temps réel et surveilleront ainsi l'état des différents équipements entrant dans le processus de production.</p>
Santé et Prévention	<p>L'Internet des objets, avec les différents capteurs embarqués, soit à travers les Wearables ou des capteurs proches du corps, forment aujourd'hui une source d'information en temps réel d'une valeur inestimable pour l'innovation dans ce secteur.</p>
Smart Grid	<p>Le Smart Grid est un réseau intelligent capable de recevoir les flux bidirectionnels d'énergie, de faire interagir les producteurs et les consommateurs, de déterminer à l'avance les besoins de la consommation et d'adapter avec flexibilité la production et la consommation d'électricité. Ce réseau peut communiquer en échangeant des informations sur les flux d'énergie, gérer avec une meilleure efficacité les pics de consommation, éviter les coupures d'électricité et réduire la charge là où cela s'avère nécessaire.</p>
Identification et traçabilité	<p>La transformation digitale dans le secteur des grandes chaînes de distribution, avec le déploiement massif de la technologie RFID, qui sert à l'identification radiofréquence et qui a permis de considérer une autre option que le code barre, nécessitant une lecture en ligne directe infra-rouge.</p> <p>La chaîne de distribution, en impliquant cette technologies depuis les usines de fabrication des produits jusqu'à l'utilisateur final acheteur de ce produit, garantit une traçabilité par produit et non pas par gammes de produits (comme c'est le cas du code barre), et a permis le développement de services très innovants autour des systèmes d'inventaires, rendant ainsi le processus d'identification des produits et le système d'inventaire plus rapide et moins coûteux.</p>

Tableau 1 - Domaines d'application de l'IoT

Des applications avantageuses pas moins intéressantes peuvent être envisagées dans d'autres domaines à savoir *l'agriculture de précision*, où le principe est le même dans tous les cas: permettre un accès ubiquitaire aux informations relatives aux différents types d'objets intelligents existants dans notre environnement afin de parvenir à automatiser le contrôle et optimiser les rendements.

## 1.7. EXEMPLES REELS DE DEPLOIEMENT

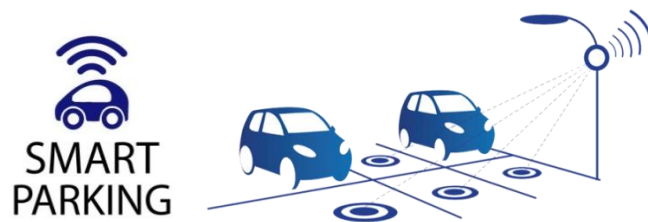
### 1.7.1. PARKING INTELLIGENT

La ville de **Santander**, en Espagne, a mis en place un réseau de capteurs communicants permettant de déployer un système global de supervision et d'information des places de parking.

Tout d'abord, les capteurs communicants étanches sont implantés au sol sur les emplacements de stationnement. Quand une voiture se gare à cet endroit, le capteur la détecte en temps réel, tout simplement en mesurant la variation du champ magnétique. Cette information est remontée périodiquement via des répéteurs sans fil vers un système de supervision, qui pourra par la suite la diffuser sur le système d'information destiné aux conducteurs, leur indiquant ainsi les emplacements libres.

Ce système d'information prend la forme de panneaux d'information, installés sur les poteaux à des endroits stratégiques de la ville. Il peut aussi alimenter une base de données consultable en ligne via une application Smartphone, qui affichera les emplacements libres sur une carte de la ville. On peut tout à fait imaginer, à terme, que les conducteurs puissent payer leur place en utilisant leur téléphone, en interagissant avec le parcmètre le plus proche.

En cas de non-paiement, le système pourrait guider la police municipale pour constater l'infraction et dresser une contravention. Le montant de la contravention dépendra du temps supplémentaire passé au-delà du temps de stationnement autorisé.



### 1.7.2. SMART GRID & METERING

ERDF (pour *Electricité Réseau Distribution France*) a mis en place le programme LINKY, le compteur électrique intelligent qui mesure les données de consommation énergétique des utilisateurs et communique en temps réel ces informations.

Un backoffice traite ces données aussi en temps réel et différents services novateurs sont alors rendu possibles : la prédiction du volume d'énergie nécessaire dans une maison ou un quartier par exemple.

Un autre aspect concerne l'intégration des différentes sources d'énergie renouvelable dans le réseau électrique, le dimensionnement intelligent de l'énergie globale, et sa distribution vers les clients finaux.



Figure 6 - Compteur électrique intelligent, par ERFD

## 2. RÉSEAUX LOW POWER WIDE AREA NETWORK

Pour les projets d'IoT, choisir le réseau de communication le plus adapté pour relier les objets connectés est une tâche très importante. Il existe des dizaines de technologies différentes, qui ont chacune leurs particularités en termes de consommation d'énergie, de prix, de portée ... etc.

Dans la section précédente, on a présenté différents technologies de communications entre les objets connectés, en se basant sur la portée. Dans le cadre de notre étude, on va s'intéresser plus spécifiquement aux réseaux à basse consommation **LPWAN**, « *Low Power Wide Area Network* ».

### 2.1. DEFINITION

Le réseau **LPWAN** ou encore **Low Power Wide Area Network** est encore peu connu, mais derrière lui se cache des technologies plus médiatisées tels que **LoRaWan** et **SigFox** [10]. Cette technologie permet d'émettre et recevoir des messages de très petites tailles, sur de très longues portées (de 5km jusqu'à 40km), avec pour avantage majeur que les composants utilisés pour émettre ces messages sont très peu coûteux et très peu énergivores (il est donc possible avec une simple batterie, d'émettre quelques messages par jours pendant 10 ans).

Pour bien identifier les spécificités et les avantages de cette technologie, voici un comparatif de **LPWAN** avec deux autres technologies communicantes couramment utilisées : *Les réseaux GSM* (3G, 4G, 5G) et *ZigBee*.

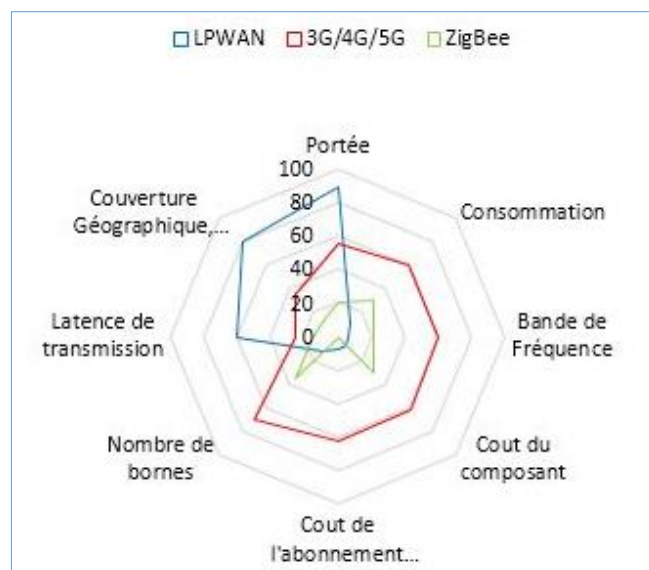


Figure 7 - Comparaison de LPWAN, réseau cellulaires mobiles et ZigBee

On peut noter que **LPWAN** se distingue sur les aspects suivants :

- ✓ Coût de mise en œuvre : *bas coût*
- ✓ Consommation : *basse consommation*
- ✓ Couverture géographique & pénétration : *large couverture géographique*
- ✓ Portée : *longue portée*

Naturellement **LPWAN** n'est pas la solution magique. Dans certains cas d'usages, cette technologie ne sera pas adaptée. En effet, dès lors que le volume de données est très important et que les notions de temps de réponses doivent être très rapides, d'autres solutions technologies doivent être utilisées.

## 2.2. CARACTERISTIQUES

A l'aide des réseaux LPWAN, il est possible, à moindre coût, de rendre communicant différents équipements. Ci-dessous les spécifications techniques du réseau **LPWAN** :

Caractéristiques	Détails
Longue Portée	5 à 40 km en champs libre
Très faible consommation	Durée de vie de la batterie en émission jusqu'à 10 ans
Coût du chipset	Moins de 2 €
Coût de souscription au réseau	Dépend de la solution mise en place
Latence de transmission	Ce n'est pas un prérequis dans le cas de LPWAN. Les applications IoT sont typiquement insensibles aux latences.
Capacité d'une Gateway à gérer de nombreux équipements	Une Gateway LPWAN est capable de gérer des milliers d'équipement par Gateway
Couverture géographique et pénétration	Excellente couverture aussi bien dans des zones urbaines que rurales. Bonne capacité de pénétration dans les bâtiments et les sous-sols.

Tableau 2 - Caractéristiques des réseaux LPWAN

## 2.3. TECHNOLOGIES

Petit focus sur deux technologies **LPWAN** : il s'agit de **Sigfox** et **LoRa**.

**Sigfox** est un véritable opérateur, qui donne accès à son réseau déjà déployé pour faire transiter les messages d'équipements **IoT Sigfox**.

**LoRa** quant à lui, ne s'occupe que de la couche physique. Il est ainsi possible de créer son propre réseau et de rendre communicant des équipements **IoT** de façon indépendante. Avec **LoRa**, il est nécessaire de déployer ses propres antennes ou de s'appuyer sur certains opérateurs.

## 2.4. ARCHITECTURE

La typologie réseau **LPWAN** est constituée de d'antennes Gateway qui permettent de faire la correspondance entre le monde Hertzien et le monde d'Internet. Ces antennes se chargent d'émettre ou de recevoir des paquets vers les différents équipements **IoT** se trouvant à proximité (5 à 40 km).

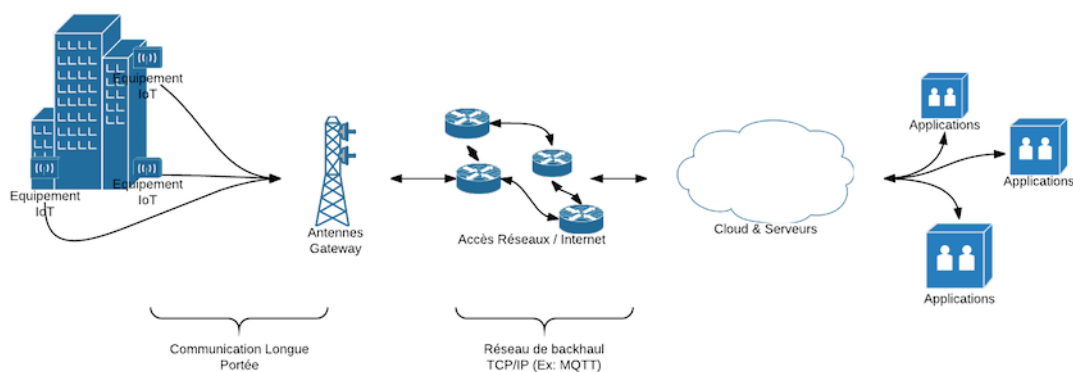


Figure 8 - Exemple d'architecture d'un réseau LPWAN

### 3. RÉSEAUX LONG RANG WIDE AREA NETWORK

#### 3.1. DEFINITION

Le protocole **LoRaWAN** est un protocole de communication pour l'IoT qui utilise une technique de modulation par étalement de spectre de type **Chirp spread spectrum** propriétaire appelée **LoRa**. La cible de **LoRaWAN** est clairement les communications longues portées à bas coût et basse consommation plutôt que les communications à débit élevé qui sont plus consommatrices en ressource CPU et en énergie [1].

En effet, les défis concernant l'interconnexion des objets résident dans leur coût, leur autonomie ainsi que leur nombre d'un point de vue réseau. Ce faible coût est obtenu par l'utilisation d'une architecture en étoile (plus simple qu'une architecture maillée), une technique de modulation plus simple à implémenter que celle des réseaux cellulaires classiques ce qui réduit le coût des composants électroniques qui lui sont dédiés ainsi que l'utilisation de bandes de fréquences libres (ne nécessitant pas de payer pour leurs utilisations). Les équipements terminaux utilisés sont majoritairement peu coûteux (1 à 5 \$) afin de pouvoir être déployés en grand nombre.

**LoRa**, pour *Long Range*, est le nom donné à la couche physique radiofréquence alors que **LoRaWAN**, pour *LoRa Wide Area Network*, désigne la couche de protocole orientée très basse consommation [11].

#### 3.2. ARCHITECTURE DU RESEAU

Le réseau LoRaWAN définit 3 typologies de composants :

- Classe A : nœud de fin de réseau, les capteurs par exemple
- Classe B : passerelle
- Classe C : serveur de données

LoRaWAN se base sur le protocole nommé LoRaMAC qui définit l'interaction entre les nœuds et les passerelles. Le protocole LoRaMAC propose plusieurs mécanismes intéressants comme la synchronisation temporelle des nœuds, la gestion d'adaptation de la puissance d'émission du nœud au travers d'un échange avec la passerelle, ainsi qu'un ensemble de clés d'identification du nœud, du réseau et de l'application [12].

Les nœuds (*classe A*) échangent des données avec les passerelles (*classe B*) en utilisant la couche radio **LoRa** et le protocole **LoRaMAC**. Les passerelles sont connectées à Internet ou sur un réseau privé en 3G, Ethernet, Wifi ou autres. Elles regroupent les messages et les transfèrent sur le serveur de données (*classe C*).

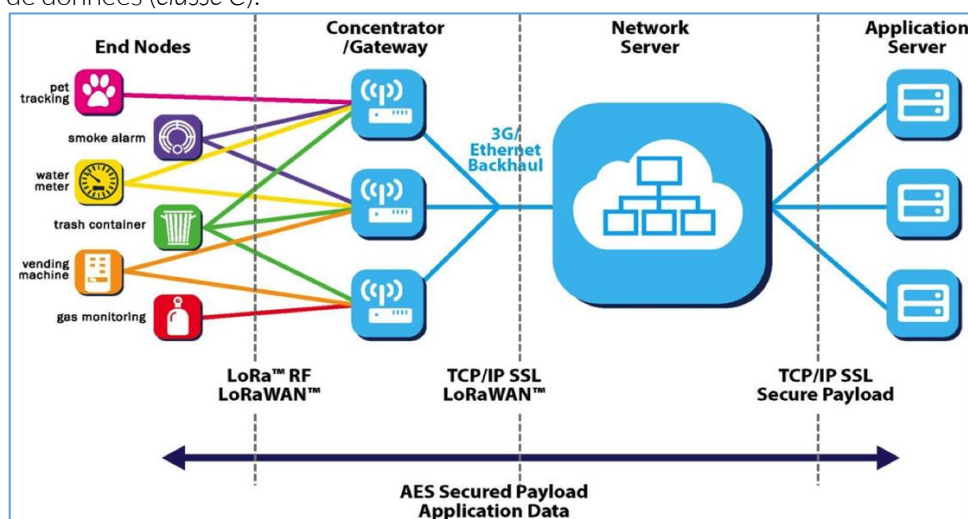


Figure 9 - Architecture d'un réseau LoRaWAN



### 3.3. CONCEPT DE SECURITE

Au sein des réseaux IoT le défi le plus marquant est la sécurité. L'analyse conceptuelle de la sécurité IoT pour la prévention contre différentes attaques et menaces de sécurité, ainsi que l'authentification et la protection de la vie privée, est réalisée par différentes activités de recherche antérieures [13].

Les problématiques générales pour les réseaux IoT sont : l'authenticité des données, la confidentialité des données, l'intégrité des données et la disponibilité du réseau [14].

Un système utilisant des objets connectés met en œuvre de nombreux protocoles. La sécurité est quelque chose qui doit se faire de bout en bout et donc se penser à tous les niveaux du modèle OSI. C'est pourquoi, les questions de sécurité et de confidentialité des données transmises doivent aussi s'imaginer dès la conception au niveau des couches PHY/MAC. Enfin, l'essor de la radio logicielle et de plateforme à bas coût offre au cybercriminel un accès beaucoup plus aisé aux couches basses des réseaux de communication ce qui risque d'accroître les menaces sur un système d'information.

Dans [15], le rapport sur l'état des cyber-menaces en 2017, Symantec déclare qu'en un an le nombre d'attaques visant les objets connectés a été multiplié par 8. En effet, si 6 000 attaques IoT avaient été recensées en 2016, ce sont bien 50 000 attaques qui l'ont été en 2017. Majoritairement, c'est le service Telnet (50,5%) qui a été ciblé mais le HTTP (32,4%) n'est pas en reste.

Un large éventail de cyber-attaques comme les attaques par déni de service (DoS), les attaques par déni de service distribuées (DDoS), les attaques réfléchies (RDoS) ainsi que les attaques réfléchies distribuées (DRDoS) ciblent spécialement les réseaux LPWAN. Ces derniers qui s'appuient sur des fréquences libres, et donc écoutables par tout le monde.

La technologie LoRaWAN, en effet, prend soin de chiffrer par défaut tous les messages de bout en bout, depuis l'objet connecté au serveur applicatif. Tous les messages sont par ailleurs signés entre l'objet connecté et les serveurs du réseau LoRaWAN. Ces opérations s'appuient sur deux clés différentes : **AppSKey** pour le chiffrement, et **NwkSKey** pour la signature.

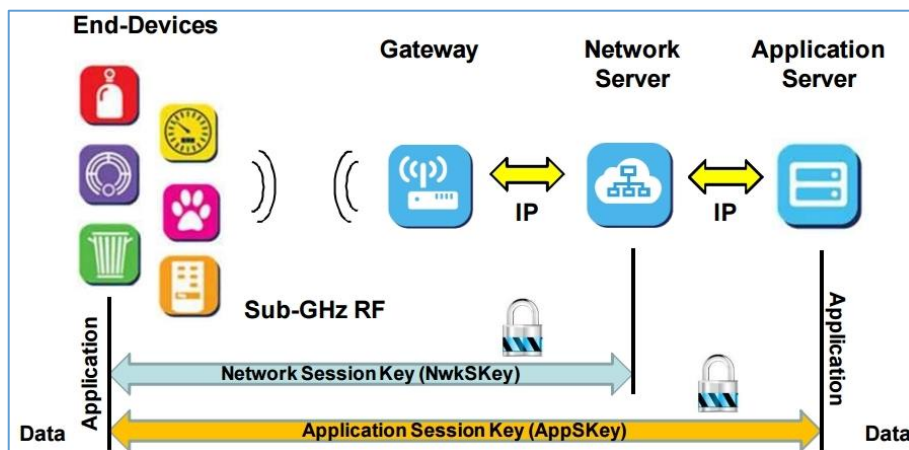


Figure 10 - Le processus de sécurité dans les réseaux LoRaWAN

Malheureusement, la manière dont ce chiffrement est réalisé n'est pas optimale et ouvre la porte à des attaques de déchiffrement partiel ou total.

Avec Sigfox, la sécurité dans la connexion au réseau est un point très important pour la société qui dit avoir mis en place des protections contre l'anti-rejeu, des messages de brouillage, ou le séquençage par exemple.

Enfin la sécurité de Sigfox se limite aux transmissions radio, contrairement au LoRa qui se focalise sur la sécurité des communications réseau.

### 3.4. EXEMPLES D'UTILISATION REELS

#### 3.4.1. SMART CITY

Une expérimentation d'un réseau privé **LoRa** a été réalisée dans un bâtiment (19 étages) dans le nord de l'**Italie**. L'objectif de son installation est de surveiller et de contrôler la température et l'humidité de différentes pièces, dans le but de réduire les coûts liés au chauffage, à la ventilation et à la climatisation.

**Marco Centaro** remonte en 2016 que l'installation comprenant une passerelle, 32 capteurs et le serveur de collecte est toujours opérationnelle et est considérée comme la solution technologique la plus adaptée pour plusieurs autres bâtiments.

Une autre étude a été réalisée par **Fanghao Yu** en 2017 sur le déploiement d'un réseau LoRaWAN pour la région du **Grand Londres** permettant de remonter la qualité de l'air et la surveillance de la congestion du trafic routier. Le design du réseau remonte la nécessité d'installer 19 petites cellules pour le centre de Londres et 28 plus grandes cellules pour la grande périphérie de Londres. Un total de 11681 terminaux sont inclus dans les 47 cellules hexagonales.

#### 3.4.2. MONITORING INDUSTRIEL

La simulation de **Kurtoglu** en 2017 montre que LoRaWAN a un avantage énergétique important par rapport à ZigBee pour les réseaux de capteurs sans fil linéaires à longue portée, comme cela serait nécessaire pour surveiller certains types d'infrastructures telles que les lignes de transport et les pipelines. De plus, l'énergie requise pour maintenir le point de consommation d'énergie le plus élevé du réseau, moins de 6 Joules par jour, est suffisamment faible pour rendre possible l'alimentation du réseau proposé en utilisant des sources d'énergie renouvelables telles que l'énergie solaire.

L'évaluation en 2016 par **Petäjärvi** de LoraWAN sur un cas réel d'usage d'équipement de santé Lora montre que les paquets sont délivrés à 96,7%, sur tout le campus d'Oulo en **Finlande**. Le campus recouvre une surface de 570 m sur 320 mètres essentiellement en bâtiment intérieur. Le résultat de l'étude montre que LoRa est une technologie attractive pour la surveillance des patients, la gestion du personnel dans les hôpitaux, la surveillance du bien-être du personnel sur le lieu de travail, ainsi que le suivi de la santé et la sécurité des personnes en extérieur.

#### 3.4.3. AGRICULTURE

L'étude menée par **Bellini** sur l'activité des bovins permet de connaître leur température et donc leur état de santé, avec un capteur accéléromètre pour connaître leur activité. Chaque animal possède un collier avec un accéléromètre et une connectivité sans fil avec un équipement LoRa, en utilisant le réseau LoRaWAN. La batterie de 400mAh avec une transmission toutes les heures des informations de l'accéléromètre sur une distance de plus de 10 km à une durée de vie de 5 ans. Le coût du matériel revient à 25 dollars pour 100 unités.

## 4. L'ANALYSE DE DONNEES

### 4.1. INTRODUCTION

L'accroissement des processus métiers gérés par l'informatique (la digitalisation) a fait exploser le volume et la variété des données produites depuis la bulle internet puis mobile et objets connectés. On peut accéder à des données externes telles que les opens datas ou les réseaux sociaux.

Savoir exploiter ces données est un enjeu important : d'abord parce qu'il est essentiel de s'assurer de la pertinence de celles-ci ; d'autre part parce que leurs résultats d'analyse fournissent des informations importantes permettant d'améliorer, d'optimiser ou d'anticiper les processus métiers.

Analyse de données	
<p>L'analyse de données est un ensemble de méthodes statistiques appliquées à un jeu de données dans le but d'extraire des informations pertinentes ; on appelle cette extraction fouille de données. Le but est de dégager des tendances, des profils, de détecter des comportements, des anomalies ou de trouver des liens et des règles.</p> <p>Il existe deux grands types d'analyse de données : l'Analyse descriptive et l'Analyse prédictive.</p>	
<p><b>Analyse descriptive</b></p>	<p>L'analyse descriptive a pour but de résumer les données en leur assignant une nouvelle représentation, de synthétiser en faisant ressortir ce qui est dissimulé par le volume.</p> <p>On peut classer les individus dans des catégories, trouver les individus les plus proches ou les plus éloignés entre eux ; mais aussi trouver les exceptions ou les cas atypiques. On peut également voir si des variables sont proches, expliquer une variable en fonction des autres ou encore repérer les variables les plus influentes.</p>
<p><b>Analyse prédictive</b></p>	<p>L'analyse prédictive consiste à analyser les données actuelles afin de faire des hypothèses sur des comportements futurs. On se sert des données que l'on possède déjà pour extrapoler et deviner le comportement de nouveaux individus mais également l'évolution des individus déjà présents.</p>

Les domaines d'application sont nombreux : Banque/Finance, Assurance, Médecine, Météorologie, IoT, Sociologie, etc. L'objectif consiste, à présent, à avoir des outils qui intègrent plus de données et qui les analysent efficacement.



Figure 11 - Analyse de données collectés de l'IoT

## 4.2. APPRENTISSAGE PROFOND -DEEP LEARNING-

Les méthodes d'Apprentissage profond sont largement appliquées à divers domaines de la science et de l'ingénierie tels que la reconnaissance de la parole, les classifications d'images et les méthodes d'apprentissage dans le traitement du langage. De même, les techniques traditionnelles de traitement des données ont plusieurs limites au traitement d'une grande quantité de données [16].

En outre, l'analyse Big Data nécessite des algorithmes nouveaux et sophistiqués basés sur des techniques d'Apprentissage Automatique –Machine Learning– et d'Apprentissage profond pour traiter des données en temps réel avec une précision et une efficacité élevées. Cependant, récemment, la recherche a incorporé diverses techniques d'Apprentissage Profond avec différents mécanismes d'apprentissage et de formation pour un traitement de données à grande vitesse [16].

L'**Apprentissage Profond** (En anglais: **Deep Learning**) est une forme d'**Intelligence Artificielle**, dérivée de l'**Apprentissage Automatique** (En anglais: **Machine Learning**) qui fait appel à de nombreuses couches de neurones artificiels qui, par leur interaction, permettent aux ordinateurs d'apprendre progressivement et efficacement à partir de mégadonnées.

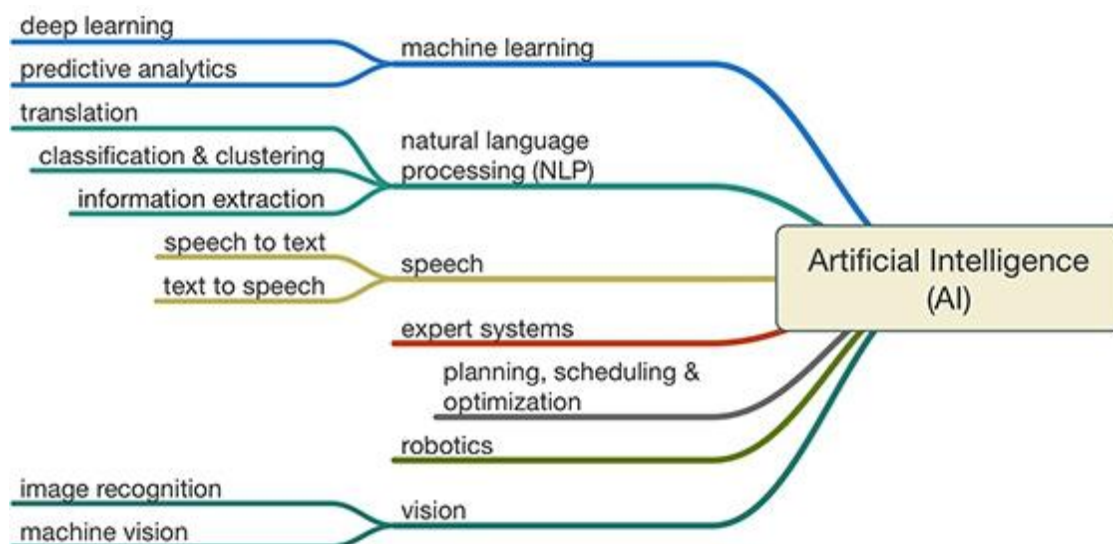


Figure 12 – Les branches de l'Intelligence Artificielle

Le Deep Learning est une technique d'apprentissage permettant d'apprendre, à partir de jeu de données, images ou textes, certains comportements afin d'arriver à associer une entrée à une sortie. C'est un domaine qui connaît actuellement un développement très important. Même si tous les problèmes ne peuvent pas être résolus avec ces méthodes, il n'en reste pas moins qu'un grand nombre de domaines ont largement évolué ces dernières années grâce à leurs apports.

La grande évolution apportée par le Deep Learning réside dans son haut niveau d'abstraction des données. En d'autres termes, contrairement aux autres méthodes de Machine Learning, le Deep Learning repose sur le postulat qu'en mettant en place un algorithme avec une architecture "assez complexe", celui-ci devrait "être capable d'apprendre seul". Pour ce fait, le Deep Learning repose sur un modèle s'inspirant du cerveau humain et fondé sur des réseaux de neurones [17].

L'**Apprentissage Profond** facilite l'acquisition de règles complexes et est largement utilisé dans *la reconnaissance d'images, la reconnaissance de la parole et des formes ainsi qu'à la Robotique et la Vision Artificielle*.

Le Deep Learning se base sur les réseaux de neurones profonds qui visent à mimer la profondeur des couches d'un cerveau humain ; le cerveau humain est « profond », dans le sens où chaque action est le résultat d'une longue chaîne de communications synaptiques avec de nombreuses couches de traitement. Le deep learning réunit une classe d'algorithmes d'apprentissage correspondants à ces architectures profondes. Il est souvent utilisé pour un apprentissage « de bout en bout », c'est à dire l'apprentissage simultané des caractéristiques utiles des données, et de la meilleure façon de les utiliser.

Parmi les exemples récents les plus remarquables : le programme de reconnaissance des visages de Facebook, la victoire d'AlphaGo ou encore les outils d'aide à la conduite assistée et autonome (ADAS), la santé avec la recherche de cellules cancéreuses par la start-up DreamQuark, ou la reconnaissance de parole.

### 4.3. QU'EST-CE QU'UN RESEAU DE NEURONES ?

Les réseaux de neurones ou Neural Networks sont des fonctions mathématiques à plusieurs paramètres, ajustables. L'analogie date des premiers automates proposés en 1943 par Warren McCulloch et Walter Pitts. Comme dans les neurones du cerveau où des connexions se créent, disparaissent ou se renforcent en fonction de différents stimuli et produisent une action, les réseaux de neurones artificiels (ou formels) ajustent des paramètres (baptisés poids synaptiques en référence au fonctionnement biologique du cerveau) en fonction de données d'entrée afin de fournir la meilleure réponse possible.

Comme l'apprentissage s'observe avant tout chez les êtres vivants, plusieurs techniques s'inspirent de la biologie. Du fait, ces réseaux de neurones artificiels imitent la structure du cerveau sous une forme idéalisée.

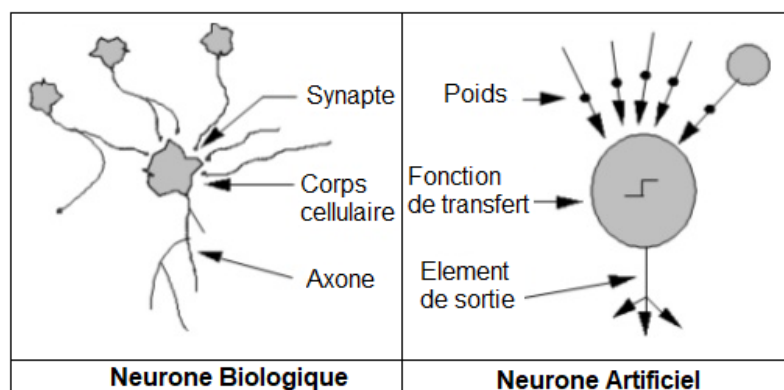


Figure 13 - La mise en correspondance Neurone Biologique / Neurone Artificiel

En machine learning, un neurone fait une combinaison linéaire des entrées qu'il reçoit, à laquelle il ajoute une valeur appelée *biais*. Une fonction non linéaire, dite d'activation, (comme par exemple tangente hyperbolique) est alors appliquée à la valeur de sortie. Cette valeur est ensuite transmise à la couche de neurone suivante. Chaque neurone effectue ainsi un calcul très rudimentaire, et c'est la succession des couches de neurones qui permet d'obtenir des réseaux complexes.

Durant cette phase dite « d'entraînement », le réseau va ajuster automatiquement les paramètres de chaque neurone, c'est-à-dire les valeurs des poids et du biais afin de minimiser l'erreur moyenne calculée sur l'ensemble des exemples entre la sortie attendue et celle observée.

L'hypothèse est qu'après cette phase d'entraînement, le réseau sera capable de traiter de manière satisfaisante de nouveaux exemples, dont la sortie est inconnue, en fonction de ce qu'il a « appris ».

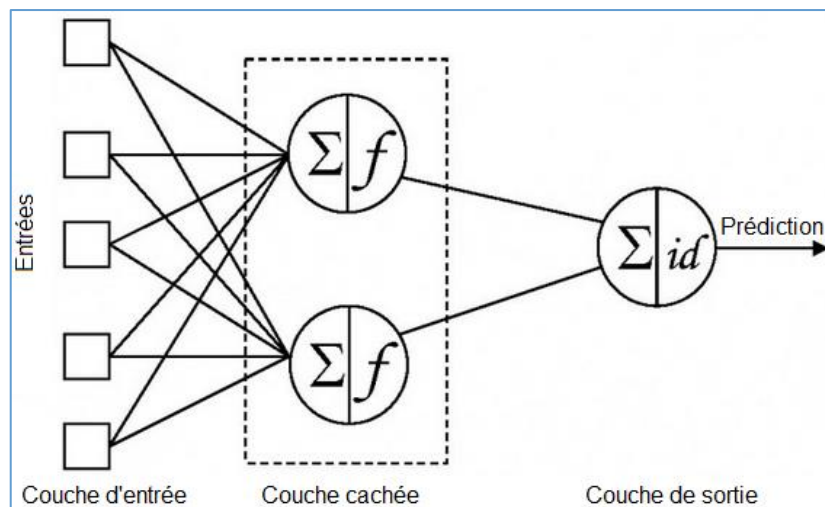


Figure 14 - Architecture générale d'un réseau de neurones

#### 4.4. ARCHITECTURES DE L'APPRENTISSAGE PROFOND

Le nombre d'architectures et d'algorithmes utilisés dans l'apprentissage profond est vaste et varié, depuis les années 90. Ces architectures sont appliquées dans un large éventail de scénarios, tels que la reconnaissance des images et vocale, ainsi que l'écriture manuscrite, etc.

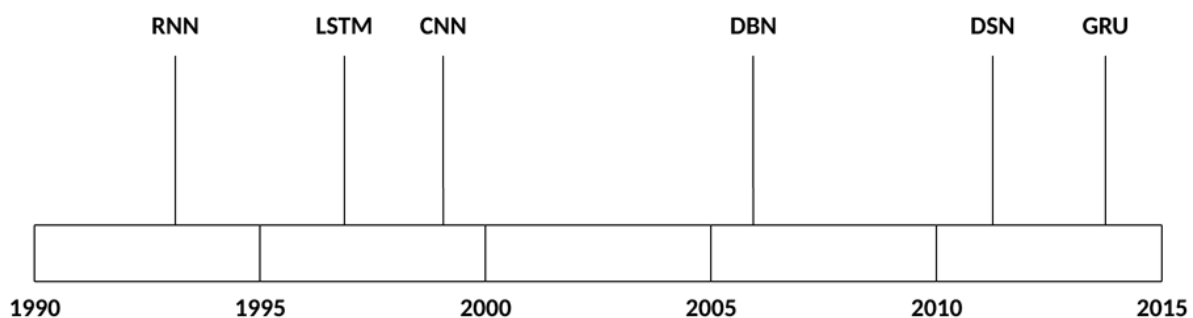


Figure 15 - Architectures de l'Apprentissage Profond de 1990 jusqu'à 2015

##### 4.4.1. RESEAUX DE NEURONES PROFONDS (DNN)

Les **réseaux de neurones multicouches** peuvent comporter des millions de neurones, répartis en plusieurs dizaines de couches. Ils sont utilisés en Apprentissage Profond pour concevoir des mécanismes d'apprentissage supervisés et non supervisés.

Dans ces architectures mathématiques, chaque neurone effectue des calculs simples mais les données d'entrées passent à travers plusieurs couches de calcul avant de produire une sortie. Les résultats de la première couche de neurones servent d'entrée au calcul de la couche suivante et ainsi de suite. Il est possible de jouer sur les différents paramètres de l'architecture du réseau : le nombre de couches, le type de chaque couche, le nombre de neurones qui composent chaque couche...etc.

##### 4.4.2. RESEAUX DE NEURONES RECURRENTS (RNN)

Les **réseaux de neurones récurrents** sont une extension des **réseaux de neurones** traditionnels, qui ont été introduits dès les années 50 - le perceptron est présenté en 1958. Cependant ils sont tombés en désuétude dans les années 70 car trop coûteux en calculs. Ils redeviennent un sujet d'intérêt dans les années 90 et connaissent actuellement leur heure de gloire grâce aux architectures profondes tels que le Deep Learning et à la puissance de calcul proposée par les GPU.

Les **réseaux de neurones récurrents** sont des modèles aptes à apprendre et à générer des séquences temporelles [18]. Ces réseaux se déclinent en plusieurs variantes dont les deux principales sont : **Vanilla** et **LSTM** [19].

#### 4.4.3. LONG SHORT-TERM MEMORY (LSTM)

Parmi les différents algorithmes d'apprentissage profond, RNN a été couramment utilisé dans les applications de prévision. Comme dans [20] l'utilisation de la mémoire à long terme (LSTM) dans la génération de texte et prédisait une sortie à chaque pas de temps.

On peut décrire le fonctionnement des réseaux de neurones comme étant **un système de vote** : dans ces réseaux de neurones, il y a un processus complexe de vote dans lequel chaque paramètre passé en entrée du réseau participe (*différentes données météorologique, par exemple*).

Dans les réseaux de neurones récurrents, la prédiction est toujours **recyclée** : la ligne en pointillée dans le schéma suivant signifie que la valeur est conservée pour un jour et réutilisée le lendemain pour devenir notre prédiction de demain. A partir de cela, nous pouvons voir comment, lorsque nous manquons d'information (*s'il y a un manque d'informations météorologique pendant des heures voire même quelques jours*), nous pouvons malgré tout effectuer une bonne prédiction.

#### 4.4.4. GATED RECURRENT UNIT (GRU)

Un réseau **Gated Recurrent Unit (GRU)** est une variante des LSTM introduite en 2014. Les réseaux GRU ont des performances comparables aux LSTM pour la prédiction de séries temporelles (ex : partitions musicales, données de parole, etc). Une unité requiert moins de paramètres à apprendre qu'une unité LSTM. Un neurone n'est associé plus qu'à un état caché et les portes d'entrée et d'oubli de l'état caché sont fusionnées (*update gate*). La porte de sortie est remplacée par une porte de réinitialisation (*reset gate*).

#### a. DESCRIPTION DES ETAPES DU FONCTIONNEMENT DES RNN & LSTM

Le fonctionnement des réseaux de neurones récurrents et LSTM est représenté dans le schéma suivant :

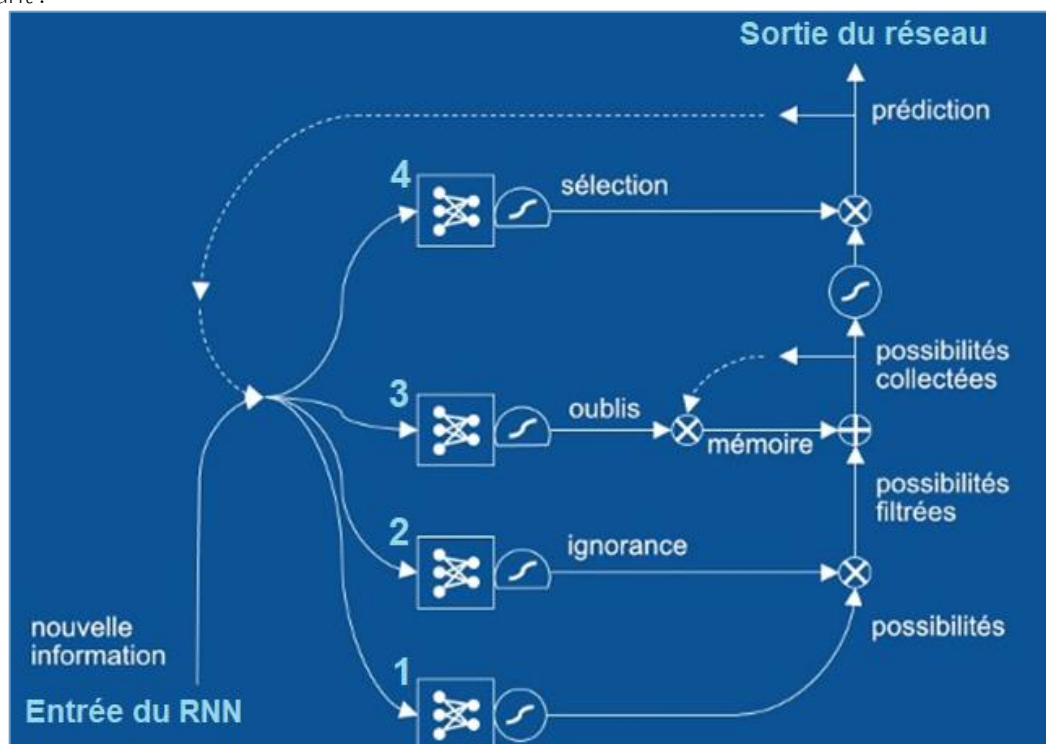


Figure 16 - Fonctionnement des RNN & LSTM



Les nouvelles informations qui arrivent au réseau passent par 4 étapes successives :

1. En entrée de ce réseau de neurones, vous passez un ensemble de paramètres tels que la température, l'humidité, la pression atmosphérique et d'autres.

Ces paramètres passent dans un premier temps, par une **fonction d'écrasement** (représentée dans le schéma avec un cercle) ; c'est la **tangente hyperbolique** qui aide le réseau à bien se comporter. L'idée en bref est de passer l'ensemble des résultats des votes réalisés à cette fonction qui les "écrasent".

A cette première étape nous créons beaucoup de **possibilités**, puis nous collectons celles qui ont de la mémoire au fil du temps et, de toutes ces prédictions possibles à chaque étape, nous n'en sélectionnons que quelques-unes comme prédiction pour cet instant. Chacune de ces actions (quand oublier et quand utiliser notre mémoire) sont appris par un réseau de neurones indépendant.

2. La deuxième étape permet **d'ignorer** éventuellement certaines prédictions — i.e. certaines possibilités à mesure qu'elles entrent en jeu. Il s'agit d'un mécanisme d'intention. Cela permet à des éléments qui ne sont pas immédiatement pertinents d'être mis de côté afin qu'ils *ne nuisent pas* aux prédictions à venir. Une nouvelle fois, ce mécanisme a son propre réseau de neurones, sa propre logistique et sa propre activité de gating.

On obtient alors **des possibilités filtrées**.

3. La troisième étape ajoute **la notion de mémoire**. En effet, nous voulons pouvoir nous rappeler facilement de ce qu'il s'est passé il y a de cela plusieurs étapes.

Pour expliquer comment cela fonctionne, il nous faut d'abord décrire quelques symboles que nous allons introduire ici :

- Le premier est une nouvelle fonction d'écrasement (celle représentée avec un fond plat),
- Un autre est représenté par le X dans le cercle,
- Le dernier est représenté par la croix dans le cercle.

Le symbole avec la croix dans le cercle représente une addition élément par élément de deux vecteurs. Et le X dans le cercle représente une multiplication élément par élément.

Donc, maintenant, nous n'avons pas seulement des prédictions, mais aussi des souvenirs que nous avons accumulés et que nous n'avons pas encore choisi d'oublier. On parle des **possibilités collectées**. Ici, nous avons aussi un réseau de neurones entièrement distinct qui apprend quand oublier, à partir de quels paramètres décider, quels éléments en particuliers on veut oublier, ...

4. Lorsque nous combinons nos prédictions avec nos souvenirs, nous ne voulons pas nécessairement intégrer à chaque fois toutes ces expériences passées dans nos nouvelles prédictions. Nous ajoutons une autre porte pour faire cette **sélection**.

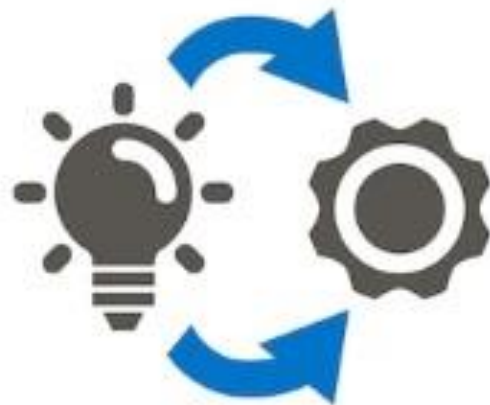
Ce filtre a son propre réseau de neurones — ainsi que son propre processus de vote — afin que nos nouvelles informations et nos prévisions précédentes puissent être utilisées pour voter sur le statut de chacune de nos portes, et que l'on puisse définir ce qui doit être maintenu en interne et ce qui doit passer en tant que prédiction. Il faut noter que nous avons aussi introduit une autre fonction d'écrasement.

Enfin, on obtient **la prédiction en sortie** du réseau de neurones récurrents avec LSTM.



## CHAPITRE *III*

### METHODOLOGIE



# 1. OUTILS MATERIELS

Pour la réalisation du prototype, on utilise la Raspberry pi 3, le module LoRa, les capteurs, un power Bank pour l'alimentation et des fils connecteurs.



Figure 17 - Les outils matériels

## 1.1. RASPBERRY PI 3

Lancé pour la première fois en 2012, la **Raspberry Pi** est un nano ordinateur à bas prix d'environ 35 \$ USD.

La nouvelle version de la **Raspberry Pi 3 Model B+** est lancée en Mars 2018, qui apporte des gains de performances à tous les étages. La nouvelle carte intègre une connexion Wi-Fi double-bande à 2,4 et 5 GHz et un débit Ethernet à 300Mbps. Un débit trois fois plus élevé que sur les anciens modèles de Raspberry. Ethernet toujours avec le support du Power over Ethernet (PoE) et le lancement futur de son propre PoE HAT. C'est une extension de carte qui permet d'alimenter son Raspberry Pi via la seule connexion Ethernet. Notons également la présence d'une connectivité Bluetooth 4.2.

Le **Raspberry Pi** offre aussi des ports GPIO permettant au technophile d'y brancher différents modules matériels. Il est donc possible d'y connecter des capteurs transformant ce petit ordinateur en plate-forme idéale pour *l'Internet des Objets*.



Figure 18 - Raspberry pi 3 Model B



Figure 19 - Raspberry pi 3 Model B+ (version 2018)

Le premier pas à faire avec une Raspberry consiste à sélectionner un système d'exploitation ainsi qu'une distribution. Voici quelques distributions populaires :

- ▶ **Raspbian**, la distribution officielle du Raspberry Pi, basé sur Debian.
- ▶ **Minibian**, une distribution pour une carte SD de faible capacité.
- ▶ **XMBC**, une distribution pour mettre en place un système multimédia maison.
- ▶ **Windows 10 IoT**.

Nous intéressant à la distribution officielle Raspbian Stretch version 29-11-2017, avec la Raspberry pi 3 model B, pour la réalisation de ce projet.

Au sein de la Raspberry, un serveur web (apache, PHP, MySQL) est mis en place.

## 1.2. CAPTEUR DE TEMPERATURE ET D'HUMIDITE

La mesure de Température et d'Humidité se fait par des capteurs de la famille **DHT\*\***. Dans cette étude on va travailler avec le **DHT11**.

Le capteur **DHT11** fournit une sortie numérique proportionnelle à la température et à l'humidité mesurées par le capteur. La technologie utilisée pour produire le capteur **DHT11** garantit une grande fiabilité, une excellente stabilité à long terme et un temps de réponse très rapide.

Les capteurs **DHT11** sont étalonnés avec précision en laboratoire. Le coefficient d'étalonnage est mémorisé dans la mémoire interne de l'OTP et cette valeur est utilisée par le processus de détection du signal interne du capteur. L'interface série monofilaire permet une intégration rapide et simple de ce capteur dans le système numérique.

L'interface physique du capteur est réalisée par un connecteur à 3 broches: +5V, GND et DATA. Les deux premières broches sont l'alimentation et la terre et elles sont utilisées pour alimenter le capteur, la troisième est le signal de sortie numérique du capteur.

Sa petite taille physique (1.05 "x0.7") et son poids très léger (seulement 0.1oz) font de cette carte un choix idéal pour la mise en œuvre de petits robots et de systèmes de surveillance d'ambiance.

Caractéristiques techniques du capteur **DHT11** :

<i>Humidité (relative %)</i>	20 ~ 80%
<i>Précision (humidité)</i>	+/- 5%
<i>Température</i>	0 ~ +50°C
<i>Précision (température)</i>	+/- 2°C
<i>Fréquence mesure max</i>	1Hz (1 mesure par seconde)
<i>Tension d'alimentation</i>	3 ~ 5 volts
<i>Stabilité à long terme</i>	+/- 1% par an

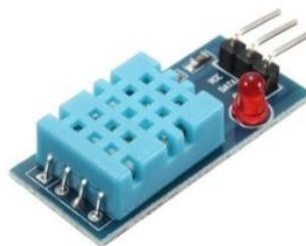


Figure 20 - Capteur de Température et d'Humidité DHT11

### 1.3. CAPTEUR DE GAZ

Le capteur analogique de gaz **MQ2** est utilisé pour la détection des fuites de gaz pour les équipements des marchés de grandes consommations et industriel. Ce capteur est conçu pour détecter le LPG (Gaz de pétrole liquéfié), i-butane, propane, méthane, alcool, hydrogène et la fumée. Il a une grande sensibilité et un temps de réponse rapide. Sa sensibilité peut d'ailleurs être ajustée par potentiomètre. Il est conçu pour un usage intérieur à température ambiante.

Le **MQ2** doit être alimenté en 5V pour le capteur physico-chimique puisse atteindre sa température de fonctionnement. Il dispose d'une sortie analogique et d'un réglage de la sensibilité par potentiomètre.



Figure 21 - Capteur de gaz MQ2

Spécifications du capteur **MQ2** :

Alimentation	5 V
Type d'Interface	Analogique
Connectique	1-Sortie 2-GND 3-VCC
Système stable à longue durée de vie	
Dimensions	40x20mm

### 1.4. WIFI LORA 32

#### 1.4.1. SPECIFICATIONS TECHNIQUES

- Support Arduino IDE, sous le nom ESP lora 32.
- 600DMIPS, 32Mbit, Tensilica LX dual core
- WIFI 802.11 b/g/n/e/i
- Bluetooth BLE
- Bande passante : 433-470 Mhz
- Sensibilité : 139dBm, (SF12, bandwidth 125khz)
- +20dBm power
- Le module Wifi LoRa 32 peut stocker jusqu'à 1044464 octets de données, c'est-à-dire environ 1 Mégaoctet (Mo).



Figure 22 - Modèle Wifi LoRa 32

Ce module possède différents entrées et sorties, comme le montre la figure suivante :

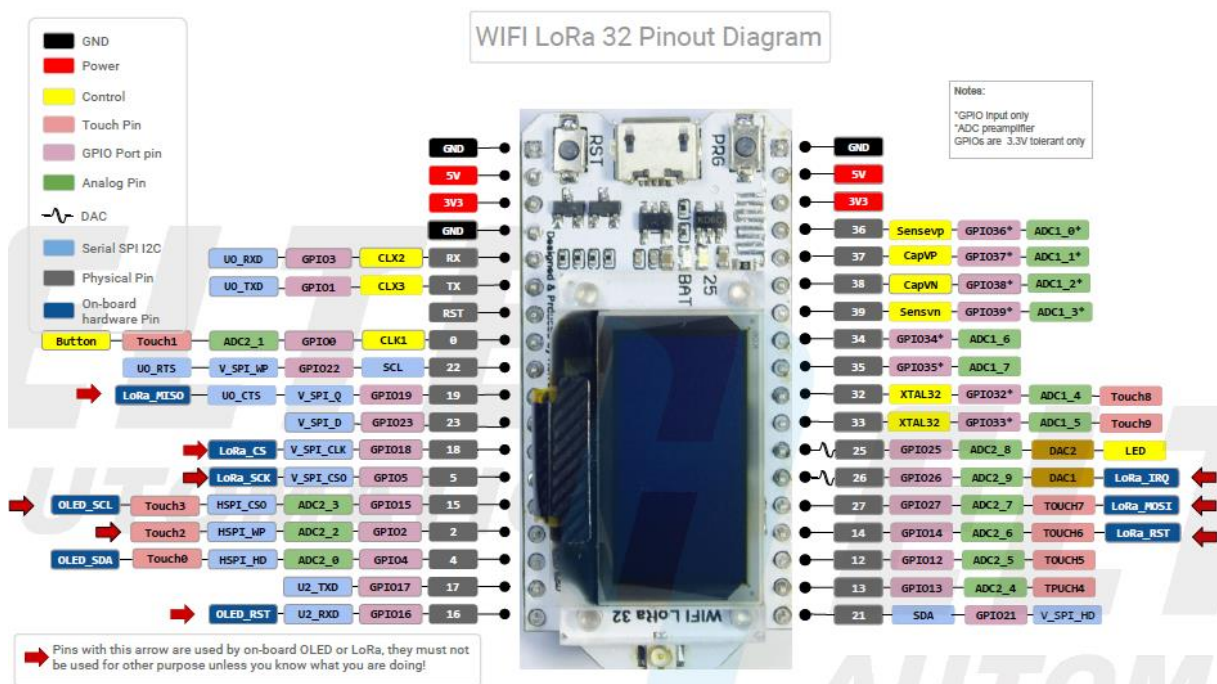


Figure 23 - Wifi LoRa 32 Pinout Diagram

Théoriquement le protocole LoRa couvre une distance de 5Km en zone urbaine et allant jusqu'à 15Km en zone rural, d'après les spécifications de LoRa Alliance pour les réseaux LoRaWAN.

Ce module de LoRa peut atteindre une distance de 2.5 à 3.5 Km, d'après des tests que des chercheurs on réaliser et partager leurs résultats sur Internet.

À notre tour, on teste la distance maximum réellement atteinte au sein de la FST de Tanger.

### 1.4.2. TESTS DE DISTANCE

- Chaque module doit avoir une adresse en hexadécimale (par exemple : 0xAA, 0xBB ... et pour la diffusion en Broadcast on utilise 0xFF)
- Le paquet envoyé par LoRa a la forme suivante :

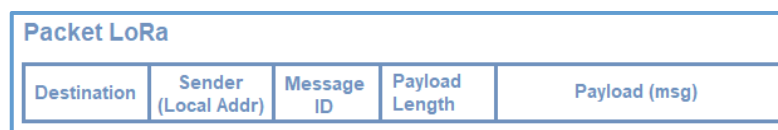
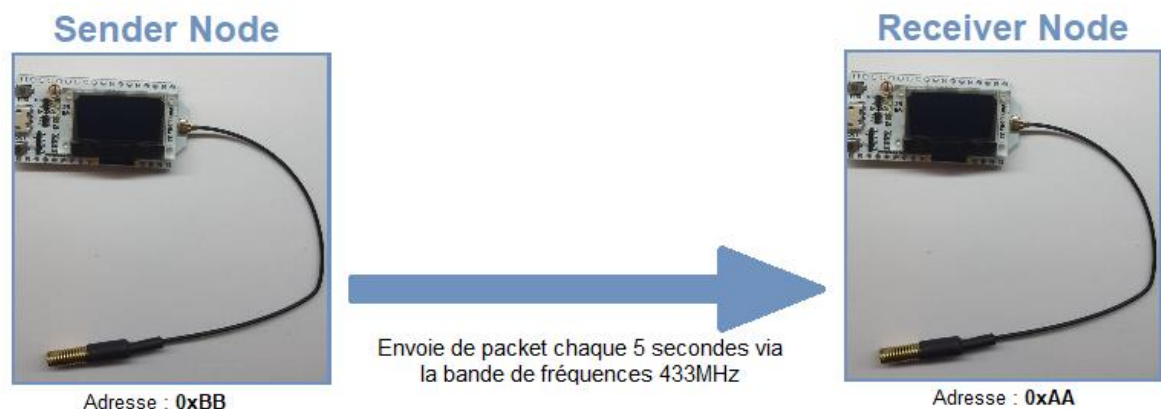


Figure 24 - Forme du paquet LoRa



Le nœud Sender envoie un paquet LoRa au nœud Receiver tout en spécifiant d'adresse de destination.

Dans notre cas les éléments du paquet seront :

- ▶ Destination : 0xAA
- ▶ Sender : 0xBB
- ▶ Message ID : valeur numérique qui s'incrémente par un à chaque envoie.
- ▶ Payload Length : La longueur du message à envoyer ; calculer par une fonction dédié.
- ▶ Payload : le message à envoyer ; les données des capteurs.

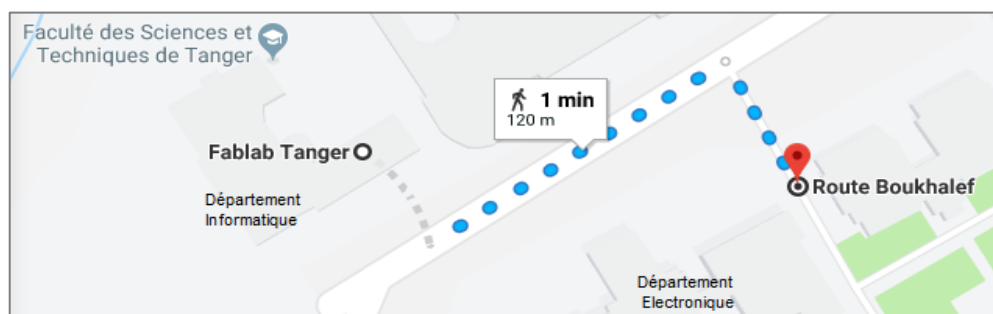
**Sender Node :** Le nœud Sender envoie le paquet LoRa élément par élément.

**Receiver Node :** Dans le nœud Receiver, le paquet doit être reçu et lu dans le même ordre qu'il a été envoyé dans le nœud Sender, pour qu'il puisse le reconstituer et le relire correctement.

### 1.4.3. DEMONSTRATION

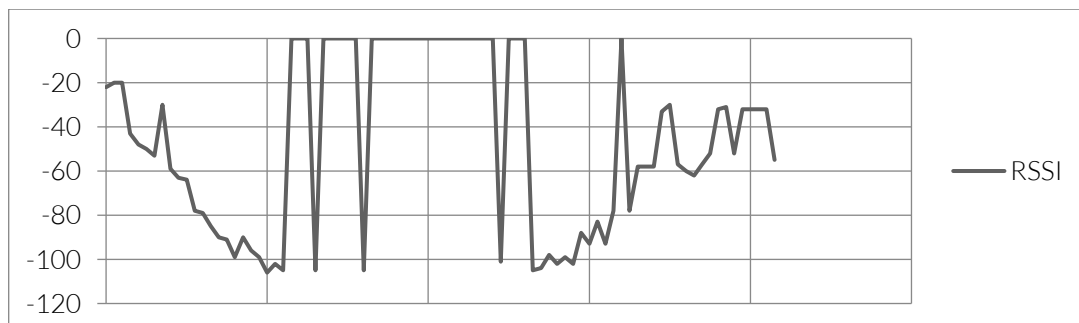
#### a. TEST 1

On a placé un nœud au laboratoire FabLab de la FSTT, et l'autre nœud est en déplacement vers le département de l'électronique (environ 100m) comme suit :



Les données analyser concerne un trajet d'aller-retour de FabLab au département de l'électronique, le diagramme suivant représente les paquets reçu en fonction du RSSI, sachant que la valeur du RSSI diminue (jusqu'à -106 dBm) avec la distance.

*RSSI=0 signifie que le paquet a été perdu, pas correctement reçu ou bien qu'un paquet inconnus est reçu au lieu du paquet attendu.*

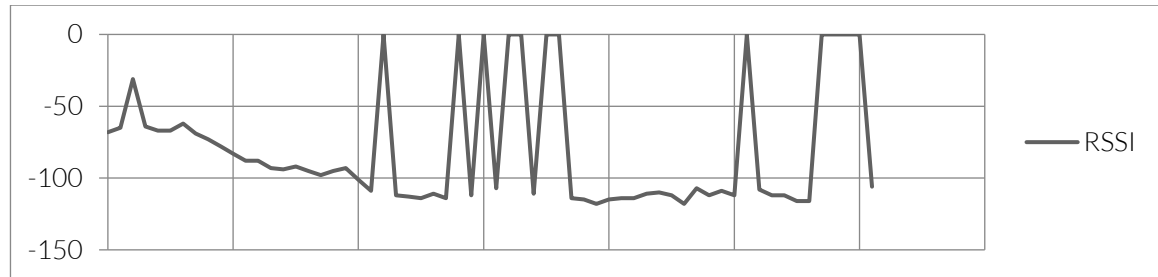
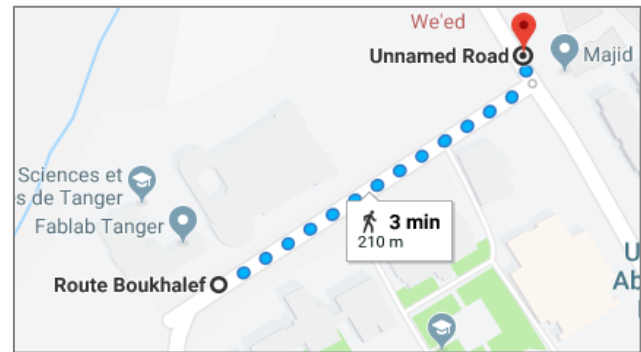


La zone où les paquets n'ont pas été reçus, il y avait plusieurs obstacles (murs).



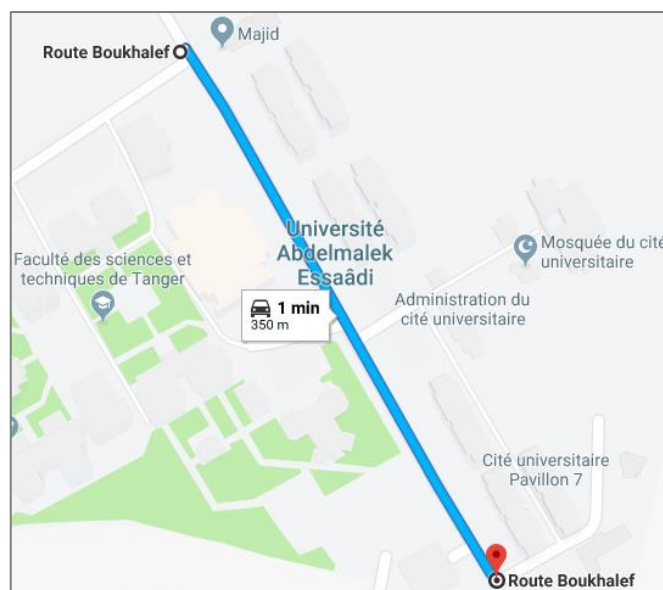
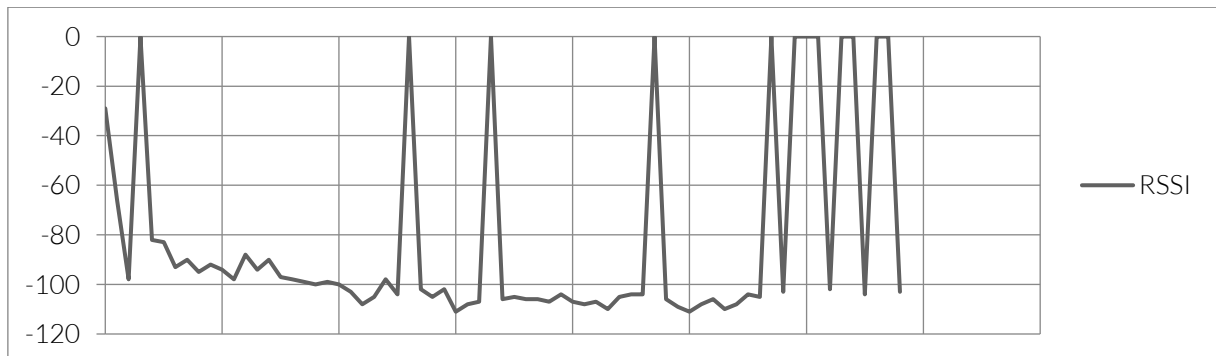
## b. TEST 2

Dans cette partie, on test une distance environ **210m** allant de FabLab jusqu'à Majid. C'est un chemin qui contient peu d'obstacles. Comme dans le test précédent, on obtient le graphe des résultats.



## c. TEST 3

Dans ce dernier, on test une distance d'environ **350m**. Comme dans les tests précédents, on obtient le graphe suivant :



Alors, la distance maximale atteinte est environ **350m** sans obstacles.

## d. RESULTAT FINAL

Finalement, la portée maximum atteinte réellement avec le module Wifi LoRa 32 est au environ de **100m** avec les obstacles (murs de bâtiments), et de **350m** sans obstacles.

## 2. FRAMEWORKS DE L'APPRENTISSAGE PROFOND

Ci-dessous, un ensemble des frameworks les plus utilisés en apprentissage profond :

TensorFlow



**TensorFlow** est un outil open source d'apprentissage automatique développé par Google sous licence Apache. Il est basé sur l'infrastructure DistBelief, et est doté d'une interface *Python*. TensorFlow est l'un des outils les plus utilisés en IA dans le domaine du Machine Learning.

TensorFlow Lite

**TensorFlow Lite** est la solution légère de TensorFlow pour les appareils mobiles (Android & iOS) et embarqués. Elle permet une inférence d'apprentissage machine sur l'appareil avec une latence et une petite taille binaire.

Keras



**Keras** est une librairie open source écrite en python et permettant d'interagir avec les algorithmes de réseaux de neurones profonds et de Machine Learning.

Caffe

Caffe

**Caffe** (Convolutional Architecture for Fast Feature Embedding) est un framework de Deep Learning à hautes performances mettant l'accent sur la vitesse, l'expression et la modularité. **Caffe** est dédié aux projets de recherche universitaire, aux prototypes des startups et aux applications industrielles à grande échelle en matière de vision par ordinateur, de traitement du langage et de création multimédia. **Caffe** est open source publié sous licence BSD.

**Caffe** est écrite en C++, et possède une interface *Python* 'pycaffe', une interface dans *MATLAB* 'matcaffe' ainsi que la ligne de commande 'cmdcaffe'.

Deeplearning4j



Le **Deeplearning4j** est le premier logiciel open-source de niveau commercial écrit en Java pour l'apprentissage automatique de réseaux de neurones profonds et l'apprentissage non supervisé.

Ce logiciel Java pour l'apprentissage automatique profonde utilise une langue spécifique au domaine (DSL) pour des reseaux de neurones artificiels qui sert a exposer leurs parametres multiples.

Le **Deeplearning4j** comprend une plateforme distribuée de l'apprentissage automatique profonde (multithread), qui fonctionne aussi sur un fil d'exécution unique (single thread).

Scikit learn



**Scikit-learn** est une bibliothèque libre Python dédiée à l'apprentissage automatique. Elle est développée par de nombreux contributeurs notamment dans le monde académique par des instituts français d'enseignement supérieur et de recherche comme *Inria* et *Télécom ParisTech*. Elle comprend notamment des fonctions pour estimer des forêts aléatoires, des régressions logistiques, des algorithmes de classification, et les machines à vecteurs de support. Elle est conçue pour s'harmoniser avec des autres bibliothèques libre Python, notamment **NumPy** et **SciPy**.

Dans la solution proposée, dans ce projet, on va utiliser **Keras** avec le Backend **TensorFlow**, et bien d'autres bibliothèques Python, qu'on va présenter dans la section suivante.



### 3. OUTILS LOGICIELS

Ici, on présente les différentes bibliothèques et outils utilisée dans la mise en place du système.

#### 3.1. ARDUINO IDE

Pour charger des programmes au module LoRa, il faut installer le support Arduino ESP32, qui sera ajouté dans la liste des Hardware :

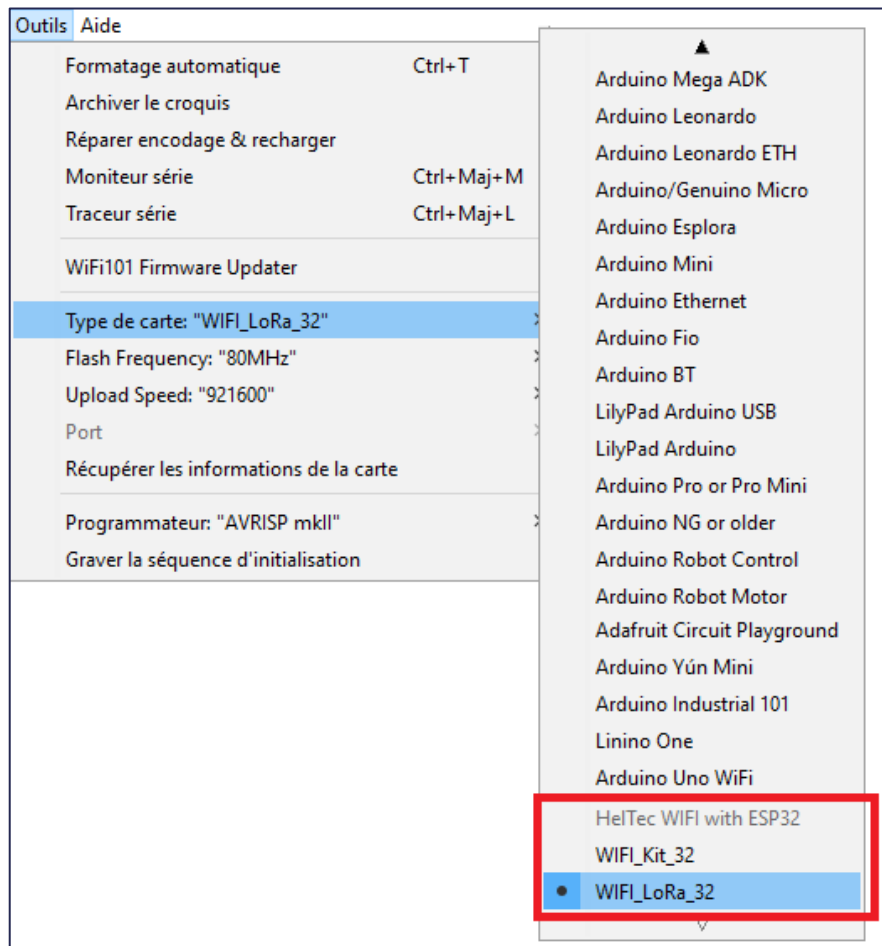


Figure 25 - Installation de Wifi LoRa 32 dans Arduino IDE

Afin de manipuler les capteurs, on aura besoin des bibliothèques dédié :

- Pour récupérer les données du capteur numérique DHT11
- Pour récupérer et calibrer le capteur analogique MQ-2, et faire la différence entre les Gaz.

Ainsi que la bibliothèque « Wifi » et « HTTPClient » sont nécessaire pour la réalisation du système.

### 3.2. BIBLIOTHEQUES PYTHON

Le modèle développer dans la deuxième partie du projet utilise les bibliothèques Python définit dans le tableau suivant :

SciPy	<b>SciPy</b> est un projet visant à unifier et fédérer un ensemble de bibliothèques Python à usage scientifique. <b>Scipy</b> utilise les tableaux et matrices du module <b>NumPy</b> . Il offre un environnement de travail scientifique très similaire à celui offert par <b>Scilab</b> , <b>GNU Octave</b> , <b>Matlab</b> voire <b>R</b> .
Keras	<b>Keras</b> est une librairie open source écrite en python et permettant d'interagir avec les algorithmes de réseaux de neurones profonds et de machine learning.
TensorFlow	<b>TensorFlow</b> est un outil open source d'apprentissage automatique développé par Google et publié sous licence Apache. <b>TensorFlow</b> est l'un des outils les plus utilisés en IA dans le domaine de Machine Learning.
Scikit-learn	<b>Scikit-learn</b> est une bibliothèque libre Python dédiée à l'apprentissage automatique. Elle comprend notamment des fonctions pour estimer des forêts aléatoires, des régressions logistiques, des algorithmes de classification, et les machines à vecteurs de support. Elle est conçue pour s'harmoniser avec des autres bibliothèques libre Python, notamment <b>NumPy</b> et <b>SciPy</b>
Pandas	<p><b>Pandas</b> est une bibliothèque utilisé pour manipuler facilement des données. Les principales fonctionnalités sont :</p> <ul style="list-style-type: none"> <li>• Récupérer des données depuis des fichiers CSV, tableaux Excel, des pages web, HDF5, etc. ;</li> <li>• Grouper, découper, alléger, déplacer, écrire les données ;</li> <li>• Ces données peuvent être à une ou deux dimensions, avec des manques, ou encore temporelles avec ou sans périodicité.</li> <li>• Traitement de données morceaux par morceaux (Intéressant dans le cas où les données peuvent dépasser la taille de la mémoire disponible).</li> </ul>
NumPy	<p><b>NumPy</b> est une extension du langage de programmation <b>Python</b>, destinée à manipuler des matrices ou tableaux multidimensionnels ainsi que des fonctions mathématiques opérant sur ces tableaux.</p> <p>Plus précisément, cette bibliothèque logicielle libre et open source fournit de multiples fonctions permettant notamment de créer directement un tableau depuis un fichier ou au contraire de sauvegarder un tableau dans un fichier, et manipuler des vecteurs, matrices et polynômes.</p>
Matplotlib	<p><b>Matplotlib</b> est une bibliothèque du langage de programmation <b>Python</b> destinée à tracer et visualiser des données sous formes de graphiques. Elle peut être combinée avec les bibliothèques <b>python</b> de calcul scientifique <b>NumPy</b> et <b>SciPy</b>. <b>Matplotlib</b> est distribuée librement et gratuitement sous une licence de style BSD.</p> <p>Plusieurs points rendent cette bibliothèque intéressante :</p> <ul style="list-style-type: none"> <li>• Export possible en de nombreux formats matriciels (PNG, JPEG...) et vectoriels (PDF, SVG...)</li> <li>• Documentation en ligne en quantité, nombreux exemples disponibles sur internet</li> <li>• Forte communauté très active</li> <li>• Interface pylab : reproduit fidèlement la syntaxe MATLAB</li> <li>• Bibliothèque haut niveau : idéale pour le calcul interactif</li> </ul>

## CHAPITRE *IV*

### IMPLEMENTATION



# 1. LORA & CAPTEURS

Dans cette première partie, on va présenter la réalisation de deux stations A & B, dans le cadre de l'Internet des Objets, capables de récupérer les données des capteurs, les envoyer par le protocole LoRa et les stocker dans la Raspberry Pi.

Le schéma général du système est le suivant :

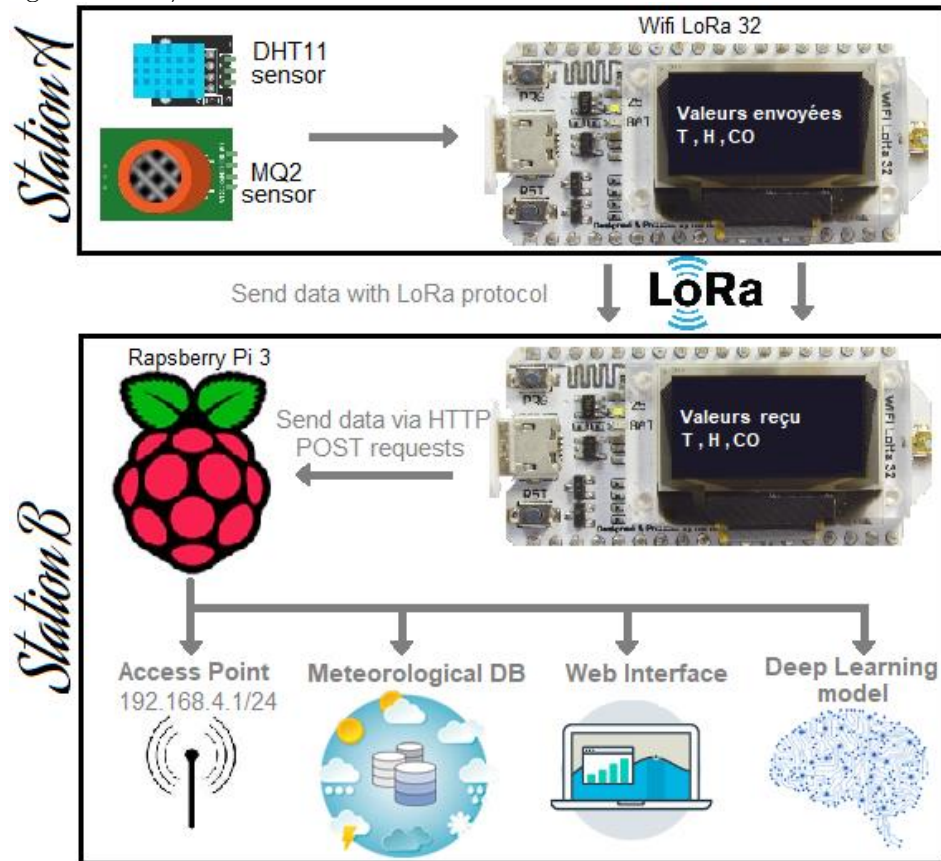


Figure 26 - Schéma générale du système IoT

La figure suivante montre le montage électrique des capteurs avec le module LoRa :

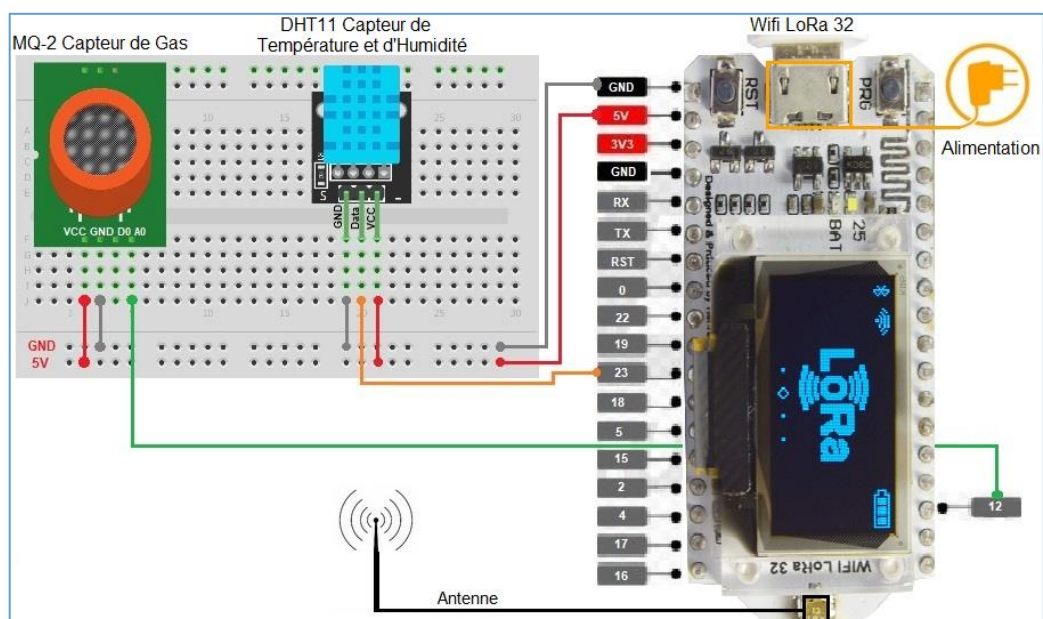


Figure 27 - Montage électrique des capteurs avec le module LoRa

## 1.1. LE PROTOTYPE REALISE

Les figures suivantes montrent le prototype des stations IoT réellement réalisé.

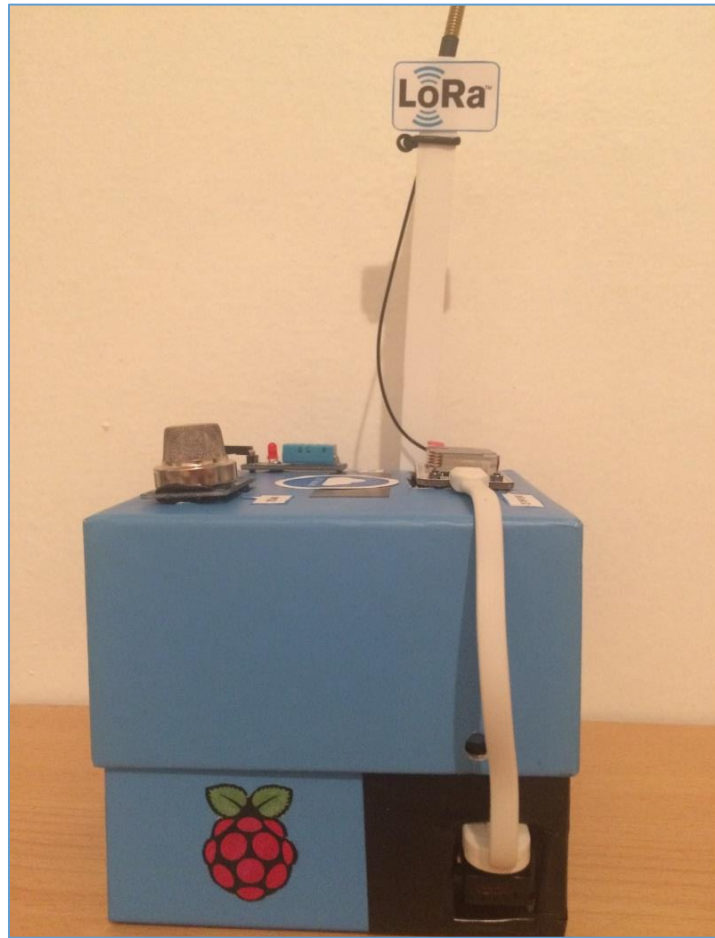


Figure 28 - Le prototype de la station IoT

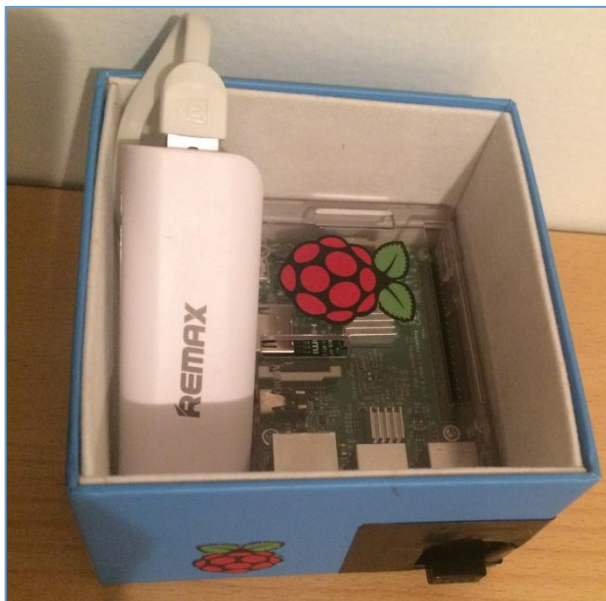


Figure 29 - Prototype, station B

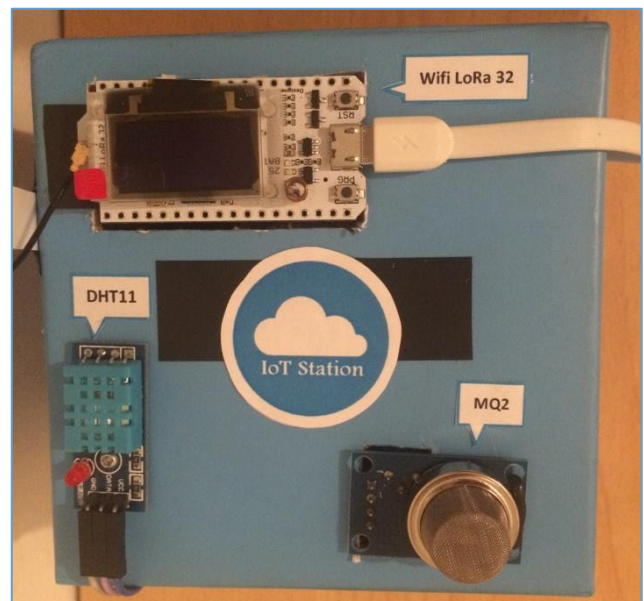


Figure 30 - Prototype, station A

## 1.2. FONCTIONNEMENT DU SYSTEME

### Station A

1. La carte LoRa branché au Capteurs (DHT11 & MQ-2) permet de récupérer différents données en temps réel ; la température ambiante (en °C), l'humidité et différents Gaz tel que le monoxyde de Carbone.
2. Les données des capteurs sont d'abord affichées dans l'écran de la carte. Et par la suite, ils sont envoyés par le protocole LoRa à l'autre carte de la station B.  
Les données envoyées sont : la température, l'humidité et la valeur du monoxyde de carbone.

### Station B

3. La carte LoRa se connecte au point d'accès RPi, et obtient une adresse IP.  
Pour se connecter à l'Internet via la Raspberry Pi, on ajoute une carte Wifi USB supplémentaire.
4. La carte LoRa reçoit les données de la station A.
5. Ces données seront envoyées par Wifi au serveur web dans la Raspberry via des requêtes http en POST.
6. Quand la Raspberry reçoit une requête http POST, qui contient les 3 valeurs (T, H, CO). On récupère d'autres données météorologiques de Tanger en temps réel l'interface Weather de Yahoo. En combinant tous ces données avec la date et l'heure, on effectue l'enregistrement dans la base de données.

## 1.3. FONCTIONNALITES DE LA RASPBERRY

La Raspberry Pi représente l'élément le plus important du système, car il s'occupe de plusieurs tâches, cité dans le tableau suivant :

- |   |                      |
|---|----------------------|
| 1 | <b>Point d'accès</b> |
|---|----------------------|
- La Raspberry pi 3 est configuré comme un point d'accès qui permettra à la carte LoRa, ou d'autres équipements, de se connecter avec un SSID et un mot de passe.
- SSID = raspberry
  - Mot de passe = \*\*\*\*\*
- Et avec un serveur DHCP, elle est capable de donner des adresses IP à tout équipement connecté, dans la plage : 192.168.4.0/24.
- L'adresse IP de la Raspberry, et du réseau RPi est : 192.168.4.1
- |   |  |
|---|--|
| 2 | <b>Application Web en Python avec Bottle</b> |
|---|--|
- Cette application web est en écoute via le port 8090, elle est réservée pour la réception des requêtes http Post de la carte LoRa. Ces requêtes qui contiennent les valeurs des capteurs venus de la Station A.
- A la réception de ces données, un programme dédié, les combinent avec d'autres données météorologiques récupérées de l'Internet via Yahoo Weather. Et l'ensemble est sauvegardé dans une base de données MySQL.
- |   |                                     |
|---|-------------------------------------|
| 3 | <b>Prédiction de la Température</b> |
|---|-------------------------------------|
- Charger le modèle entrainer et faire des prédictions sur de nouvelles valeurs. *'Partie 2 du projet'*
- |   |   |
|---|---|
| 4 | <b>Serveur Web pour l'Interface Web</b> |
|---|---|
- Afin de visualiser les résultats, une Interface Web accessible depuis la Raspberry est un bon choix. Via un serveur web (Apache, PHP, MySQL) les données météorologiques ainsi que les prédictions sont bien organisés et présentés dans la page Web.



## 2. APPRENTISSAGE PROFOND ET PREDICTIONS

Jusqu'à présent, les données stockées dans la Raspberry Pi ne sont pas utilisés. Il est temps de les exploiter pour faire des prédictions. La prédiction ou l'analyse prédictive consiste à analyser les données actuelles afin de faire des hypothèses sur des comportements futurs.

Pour cela, on va réaliser plusieurs essais de modèle d'apprentissage profond et les appliquer sur nos propres données. On présentera dans cette partie les trois essais les plus intéressants.

### 2.1. PREDICTION DE LA TEMPERATURE SIMPLE

La prédiction de la série temporelle est un problème difficile à la fois pour encadrer et à résoudre avec l'apprentissage automatique.

Dans cette étude, nous allons développer un modèle de réseaux neuronaux – Neural Network – pour la prédiction de séries temporelles en Python en utilisant la bibliothèque d'Apprentissage Profond Keras.

#### 2.1.1. PROBLEMATIQUE

Le problème que nous allons examiner ici est le problème de prédiction de la température ambiante.

C'est un problème où, étant donné une heure (heure : minutes : secondes), la tâche consiste à prédire **la température de l'air**. Les données sont enregistrés le **13/04/2018** durant 5 minutes environ, avec 146 observations.

Voici un extrait des données :

17	14-36-12,"18"
18	14-36-13,"22"
19	14-36-18,"18"
20	14-36-19,"18"
21	14-36-21,"18"

L'ensemble des données sont récupérer depuis la base de données de Raspberry pi, de l'architecture qu'on a présenté précédemment. On récupère les données avec le format csv, sous le nom *temp.csv*.

#### 2.1.2. ARCHITECTURE DU MODELE

Il s'agit d'un réseau de neurones simple qui contient : une entrée, une couche caché avec 8 neurones et une sortie.

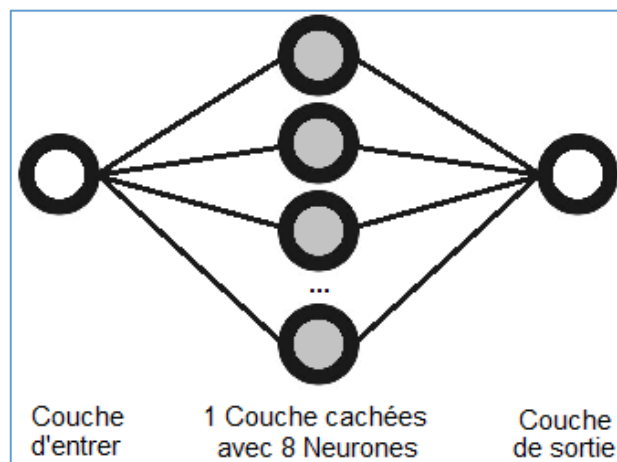


Figure 31 - Architecture du modèle de la prédiction de Température Simple

- Le principe de ce modèle est qu'on tenant en compte la température d'une heure (t), quel sera la température future (t+1) ?
- Pour en faire, on va utiliser l'environnement **SciPy** avec la librairie de l'apprentissage profond **Keras**.

### 2.1.3. DESCRIPTION DU MODELE

- Les données sont séparées en deux : données d'entraînement 70% et données de test 30%.
- On va convertir notre seule colonne de données en un ensemble de données à deux colonnes ; la première contient la température de cette seconde (t) et la seconde colonne contient la température future (t+1).
- Le modèle est constitué d'une entrée, une couche cachée avec 8 neurones et une couche en sortie.
- Dans la phase d'entraînement, il y aura 200 itérations.
- La base de données contient environ 146 lignes sous la forme :
  - « heure-minute-seconde, Température »

### 2.1.4. RESULTAT

Après l'exécution du modèle on obtient le résultat suivant :

```
Epoch 200/200
- 0s - loss: 0.6427
Train Score: 0.63 MSE (0.80 RMSE)
Test Score: 0.11 MSE (0.34 RMSE)
```

En prenant la racine carrée des scores de performance, nous pouvons voir l'erreur moyenne sur l'ensemble de données d'entraînement (Température) était de 0.81 °C (par seconde) et l'erreur moyenne sur l'ensemble de données de test était de 0.34 °C (par seconde).

Nous pouvons voir aussi, que le modèle a fait un travail qui adapte les ensembles de données d'apprentissage et de test. Il a essentiellement prédit la même valeur d'entrée que celle de sortie.

Les données sont tracées, montrant :

- En **BLEU** : les données original
- En **VERT** : les prédictions pour les données d'entraînement
- En **ROUGE** : les prédictions sur l'ensemble de données de test

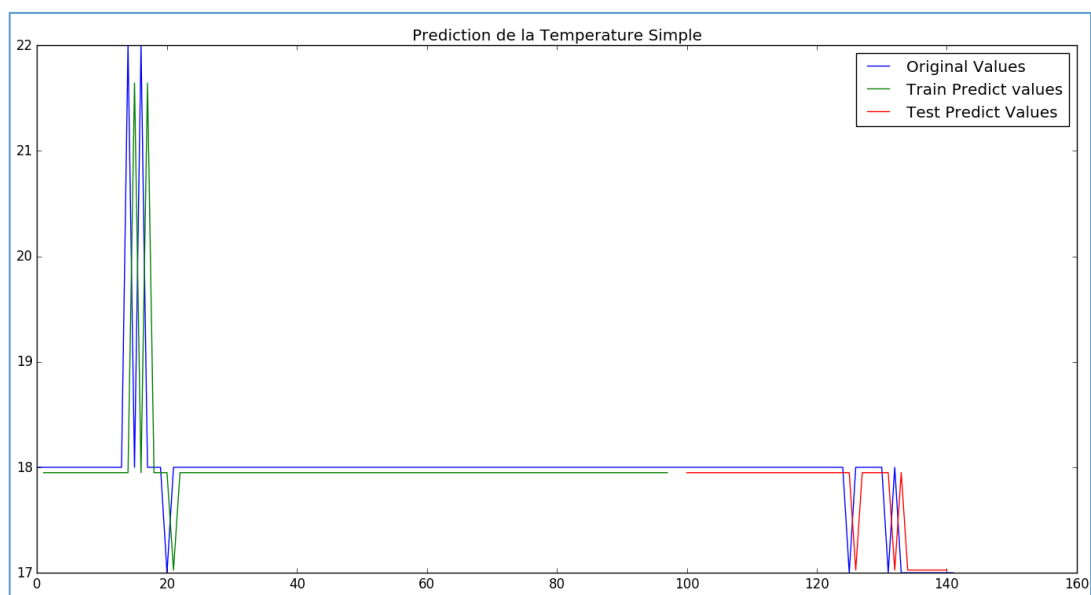


Figure 32 - Résultat de la prédiction de la Température Simple



## 2.2. PREDICTION DE LA POLLUTION DANS L'AIR

Afin d'améliorer le modèle précédent, dans ce modèle utilise un réseau de neurones récurrent avec une couche cachée en LSTM.

L'objectif de ce modèle est la prévision de la pollution de l'air en se basant sur des données trouvées dans le web.

Pour l'appliquer dans une prochaine section, sur les données de notre système.

### 2.2.1. DESCRIPTION DES DONNES

- Il s'agit d'une base de données qui rend compte de la météo et du niveau de pollution toutes les heures pendant cinq ans à Pékin, en Chine.
- Les données contiennent :
  1. Numéro de ligne
  2. Année
  3. Mois
  4. Jour
  5. Heure
  6. **Concentration de PM2.5**
  7. Point de rosée
  8. Température
  9. Pression
  10. Direction du vent
  11. Vitesse du vent cumulée
  12. Heures de neige cumulées
  13. Heures de pluie cumulées

La pollution est représentée par la concentration du PM2.5 (en  $\mu\text{g}/\text{m}^3$ ) dont une grande partie de ces particules résulte des activités polluantes comme les moteurs diesel, les suies, les particules issues de la transformation chimique des gaz polluants en nitrates et sulfates, et les déchets de la combustion.

### 2.2.2. ARCHITECTURE DU MODELE

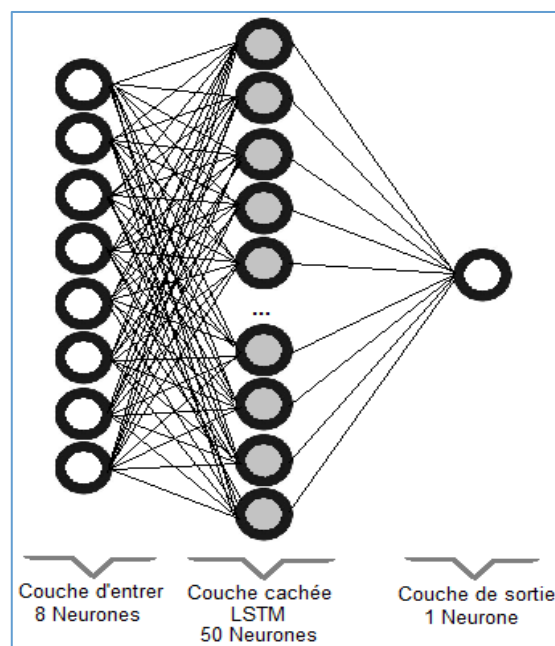


Figure 33 - Architecture du modèle de la prévision de la pollution dans l'air

### 2.2.3. APPLICATION AVEC KERAS

Ci-dessous les étapes de l'application des RNN & LSTM avec Keras en Python :

#### 1 Récupération des données

→ Téléchargement des données de l'UCI Machine learning Repository

#### 2 Préparation basic des données

- Utilisation de la librairie Pandas.
- Combiner les informations de l'heure et la date en une seule colonne.
- Eliminer la première colonne qui contient la numérotation
- On trouve des valeurs NA, il faut les remplacer par des 0.
- Eliminer les données des premières 24h parce que la valeur de PM2.5 est NA

→ Résultat : 10 colonnes qui vont être enregistré dans un nouveau fichier nommé row.scv

#### 3 Préparation des données pour LSTM

- Cela implique de cadrer l'ensemble de données comme un problème d'apprentissage **supervisé** et de **normaliser** les variables d'entrée...
  - 8 input variables (input series)
  - 1 output variable (pollution level at the current hour)

Les données résultant de l'ensemble des préparations :

1	2	3	4	5	6	7	8
pollution	dew	temp	press	wind_direction	wnd_spd	snow	rain

- Ensuite, il faut séparer les données en deux, une partie pour l'apprentissage et l'autre pour les tests du modèle :
  - Données d'apprentissage ou d'entraînement : la première année
  - Données de tests : les trois années restantes

#### 4 Définir et entrainer le modèle LSTM

- Spécifications :
  - 50 neurones dans la première couche cachée
  - 1 neurone dans la couche de sortie pour la prévision de la pollution
  - 50 itérations d'entraînement avec une taille du lot de 72
- Pour cela on utilise deux autres notions :
  - L'erreur absolue moyenne (MAE pour Mean Absolute Error): moyenne arithmétique des valeurs absolues des écarts.
  - The efficient Adam version of stochastic gradient descent

#### 5 Evaluer modèle et faire des prédictions

- Calculer la **Root Mean Squared Error (RMSE)** ; La RMSE est l'écart-type des réponses, écart-type estimé grâce à l'analyse de la variance.

### 2.2.4. RESULTAT

On remarque que durant les 50 itérations l'écart d'erreur diminue clairement, donc notre modèle apprend bien de faire les prévisions de la pollution d'après les données qu'on lui fournit en entrée du réseau.

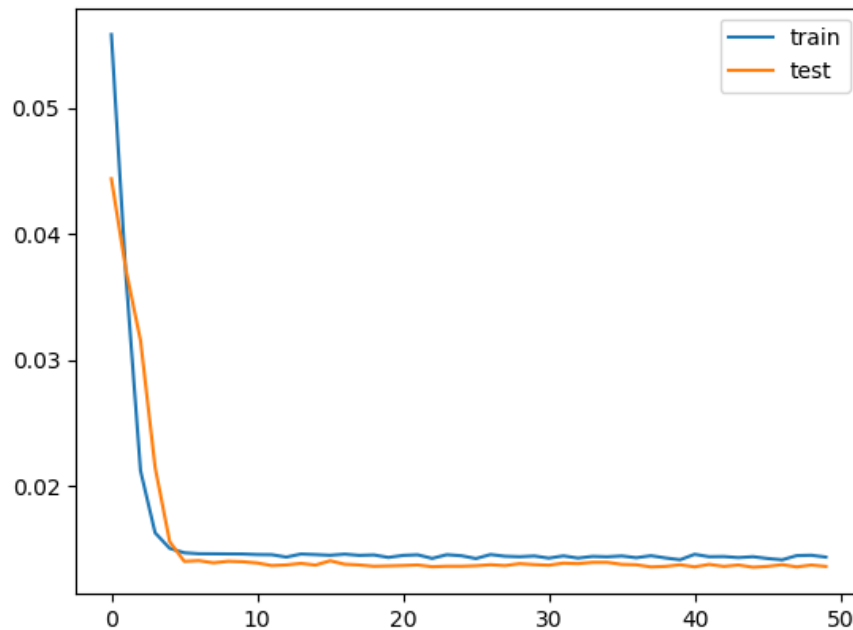


Figure 34 - La variation de l'écart d'erreur durant l'apprentissage du modèle

## 2.3. PREDICTION DE LA TEMPERATURE AVANCEE

L'objectif de cette dernière section est la réalisation d'un modèle Deep Learning plus profond capable de faire la prédiction de la Température ambiante, basé sur la variation des différentes données météorologiques récupérées depuis la Raspberry Pi de la Station B de notre système.

Ce modèle est basé sur le modèle de 'Prédiction de la pollution dans l'air', avec plus de profondeur ainsi plus de précision dans les prédictions.

### 2.3.1. DESCRIPTION DES DONNEES

La base de données du Raspberry pi contient les données suivantes :

1. Date et Heure
2. **Température**
3. Humidité
4. Monoxyde de Carbone (CO)
5. Température Extérieur
6. Pays
7. Vitesse de vent
8. Direction de vent
9. Refroidissement Eolien (ou refroidissement du vent)

### 2.3.2. ARCHITECTURE DU MODELE

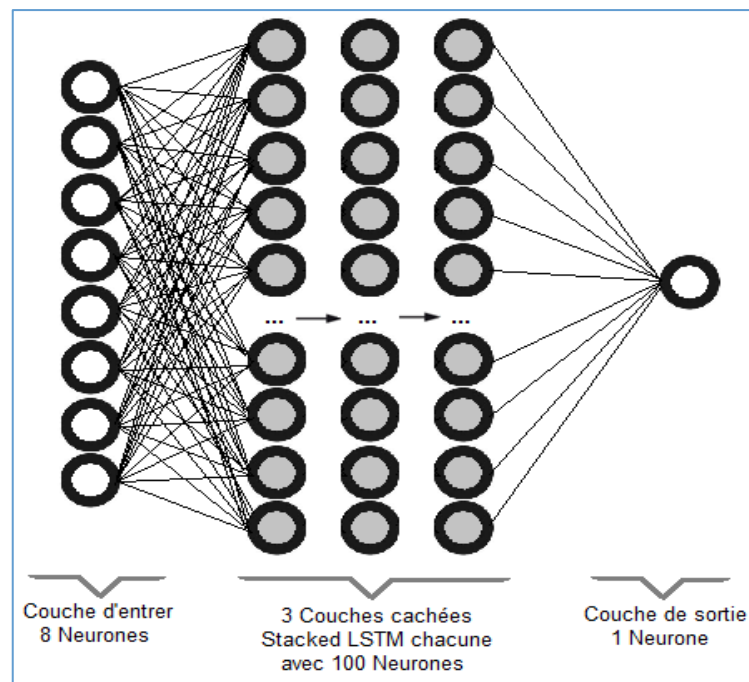


Figure 35 - Architecture du modèle de Prédiction de la température avancée

### 2.3.3. APPLICATION AVEC KERAS

#### a. TACHES REALISER SUR UN PC

##### 1 Préparation des données pour LSTM

- Cela implique de cadrer l'ensemble de données comme un problème d'apprentissage **supervisé** et de **normaliser** les variables d'entrée.
  - 8 input variables
  - 1 output variable (Température)
- Ensuite, on sépare les données en deux, une partie pour l'apprentissage et l'autre pour les tests du modèle :
  - Données d'apprentissage ou d'entraînement : 75%
  - Données de tests : 25%

##### 2 Définir, entraîner et sauvegarder le modèle

- Spécifications :
  - 1 couche d'entrée contient 8 neurones
  - 3 couches cachées LSTM contiennent chacune 100 neurones
  - 1 neurone dans la couche de sortie pour la prévision de la température
  - 350 itérations d'entraînement avec une taille du lot de 72 (Selon le cas)
    - Si on possède d'une grande base de données, il suffit de mettre 50 itérations
    - Si les données ne sont pas assez suffisantes, on augmente le nombre d'itérations ; et plus le nombre d'itérations est grand plus la précision du modèle augmente.
- Afin de sauvegarder le modèle entraîné, Keras sépare l'enregistrement en deux catégories :
  - L'enregistrement de l'architecture ou la structure du modèle avec **JSON**,
  - L'enregistrement de l'ensemble des poids du modèle avec **HDF5** format.

##### 3 Evaluer le modèle

- Calcule de RMSE.

## b. RESULTATS D'ENTRAINEMENT

On remarque que durant les 350 itérations de l'apprentissage du modèle, l'écart d'erreur diminue clairement, donc notre modèle apprend bien de faire les prévisions de la température d'après les données qu'on lui fournit en entrée du réseau.

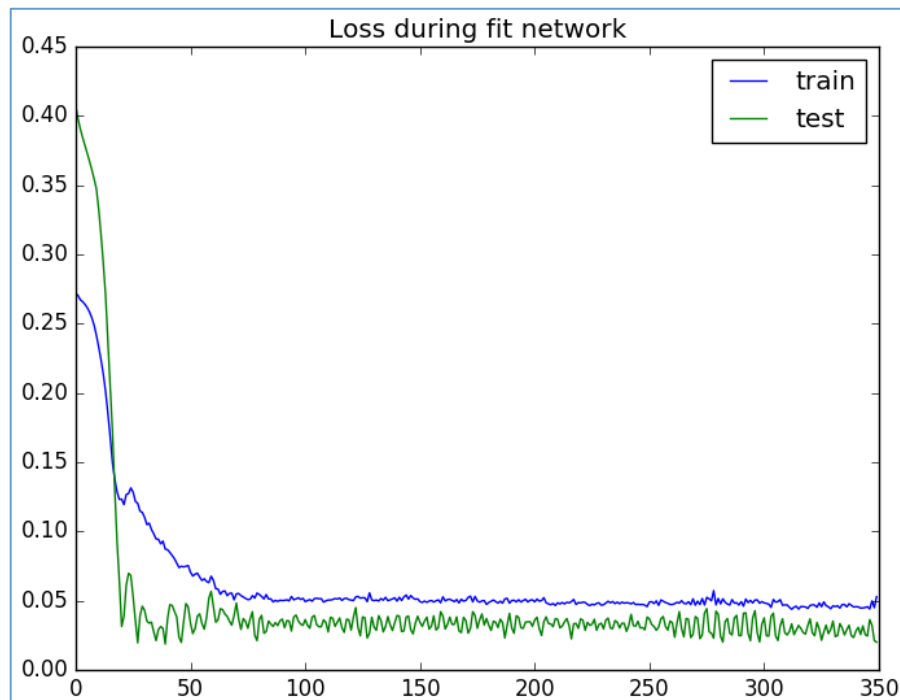


Figure 36 - La variation de l'écart d'erreur

Ainsi, avec cette architecture la valeur du RMSE atteint 0.294 durant le test du modèle.

```
Epoch 350/350
- 0s - loss: 0.0529 - val_loss: 0.0200
Saved model to disk !
Test RMSE: 0.294
```

Figure 37 - Résultat d'entrainement du modèle, valeur du RMSE

## c. TACHES REALISER SUR LA RASPBERRY PI 3

4	Préparation des nouvelles données pour LSTM
	Ici, il faut refaire l'ETAPE 1 mais pour de nouvelles données, directement importer de la RPi.
5	Charger le modèle
	On récupère les deux fichiers (JSON et HDF5) résultant de l'ETAPE 2 dans la Raspberry.
6	Evaluer le modèle et faire les prédictions
	<ul style="list-style-type: none"> <li>→ L'évaluation du modèle se fait par le calcul du RMSE.</li> <li>→ Pour la prédiction, on utilisera un nouveau jeu de données, pour s'assurer de la fiabilité et le fonctionnement de notre modèle.</li> </ul>

Les résultats sont présentés dans le Chapitre V.

### 3. VISUALISER LES DONNEES

La Raspberry pi est accessible via une adresse IP de son propre réseau. L'interface web de la Raspberry pi permet de visualiser les différents données et résultats. Le développement de cette interface est réalisé avec PHP, HTML5, CSS et JS.

Le PHP est un puissant langage permettant le développement rapide des sites web. Et, afin d'éviter les principales vulnérabilités liées aux applications web, ainsi que celles propres à la plateforme PHP, on s'oriente vers un développement sécurisé en PHP. Tout en tenant compte de différentes mesures de sécurité spécifiquement lors de l'accès à la base de données, pour minimiser les risques et améliorer la qualité de notre application.

# CHAPITRE V

## RESULTATS & DISCUSSION



# 1. RESULTAT DE LA PREDICTION

La prédiction de la température tient compte des autres données météorologique fournit en entrée du réseau. Après avoir chargé le modèle et faire les prédictions, le programme Python retourne les diagrammes suivants :

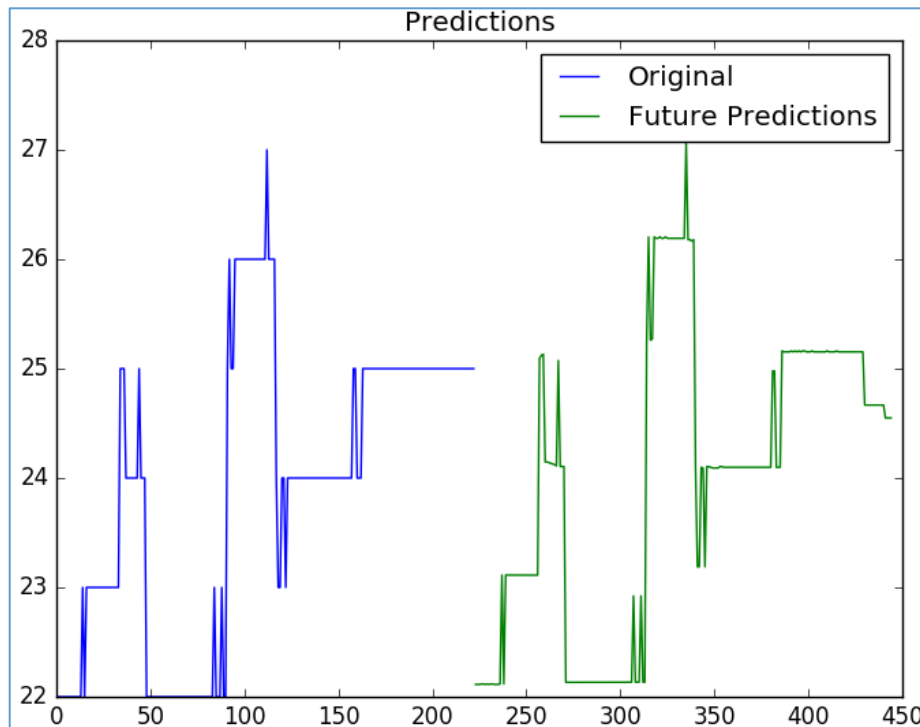


Figure 38 - La Température en entrée et les prédictions

On peut également superposés les deux courbes pour voir la différence entre la Température original et la Température prédite :

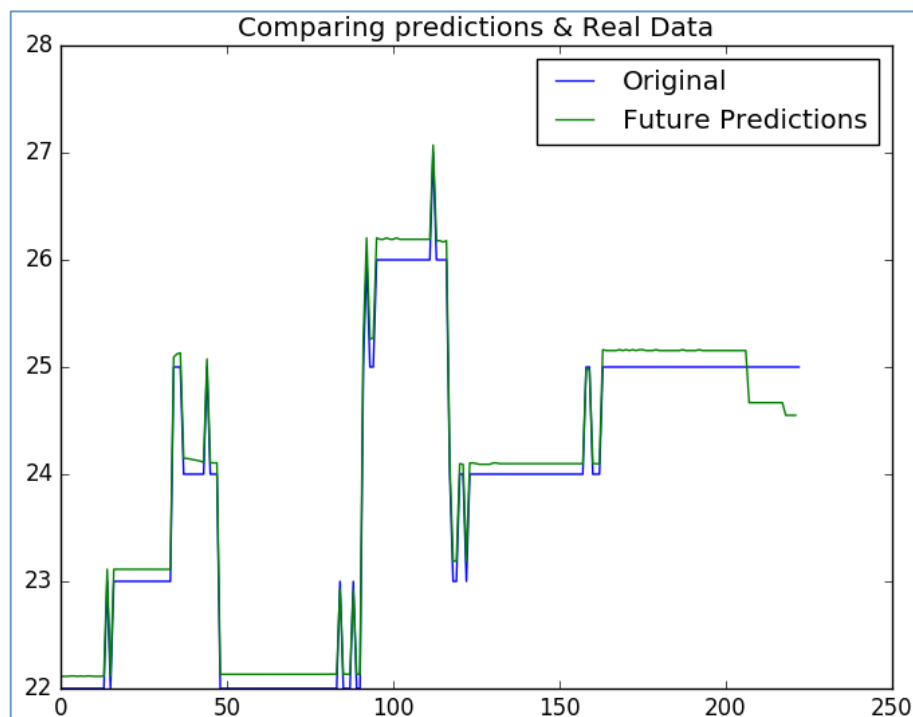


Figure 39 - Comparaison des prédictions avec les valeurs originals



## 1.1. COMPARAISONS

### 1.1.1. VARIATION DE COUCHES CACHEES

Ce dernier modèle présente la meilleure combinaison de hyper paramètres que nous avons testés. Le tableau suivant présente les résultats des autres combinaisons en ajoutant et en supprimant des couches cachées. Nous comparons la valeur RMSE dans les deux phases d'entraînement et de test.

	Nombre de couches cachées		
	1	2	3
Train RMSE	0.308	0.308	0.294
Test RMSE	0.544	0.499	0.457

Tableau 3 - Comparaison des architectures LSTM

Dans cette comparaison, on varier uniquement le nombre de couche cachée. La variation de nombre de neurones dans chaque couche caché ainsi que le nombre d'itérations, ont un grand impact sur le modèle.

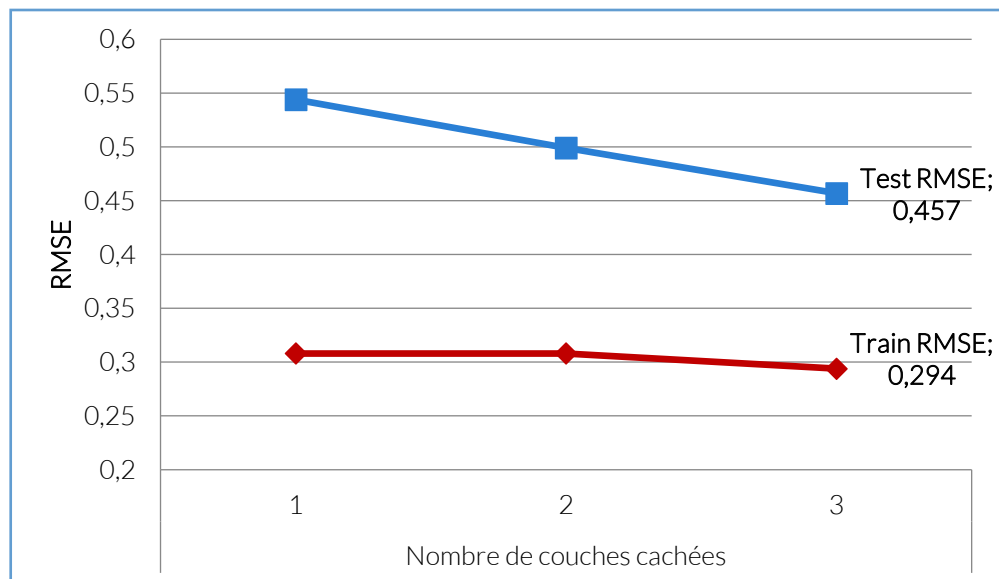


Figure 40 - RMSE par évolution de couche cachée

### 1.1.2. TAILLE DE LA BASE DE DONNEES

En comparons les résultats de la prédiction de pollution (*Chapitre 4, 2.2*) et de la prédiction de température ambiante (*Chapitre 4, 2.3*). On remarque que, quand la base de données est grande à 43800 enregistrement environ il suffit de mettre 50 epochs. Alors que si la base de données ne contient que 220 enregistrement on a besoin de 350 epochs pour avoir un résultat similaire du premier.

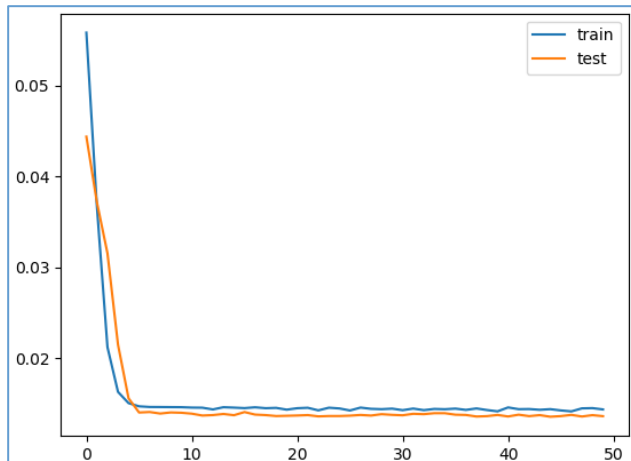


Figure 41 - Résultat de la prédiction de pollution

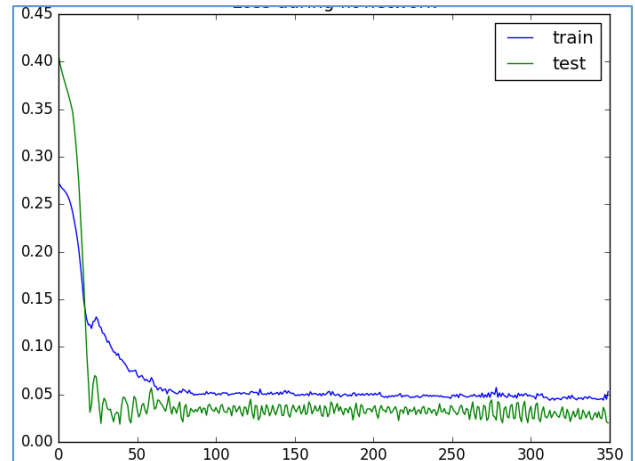


Figure 42 - Résultat de la prédiction de température

	Prédiction	
	Pollution	Température
Taille BD	43800	220
Epochs	50	350

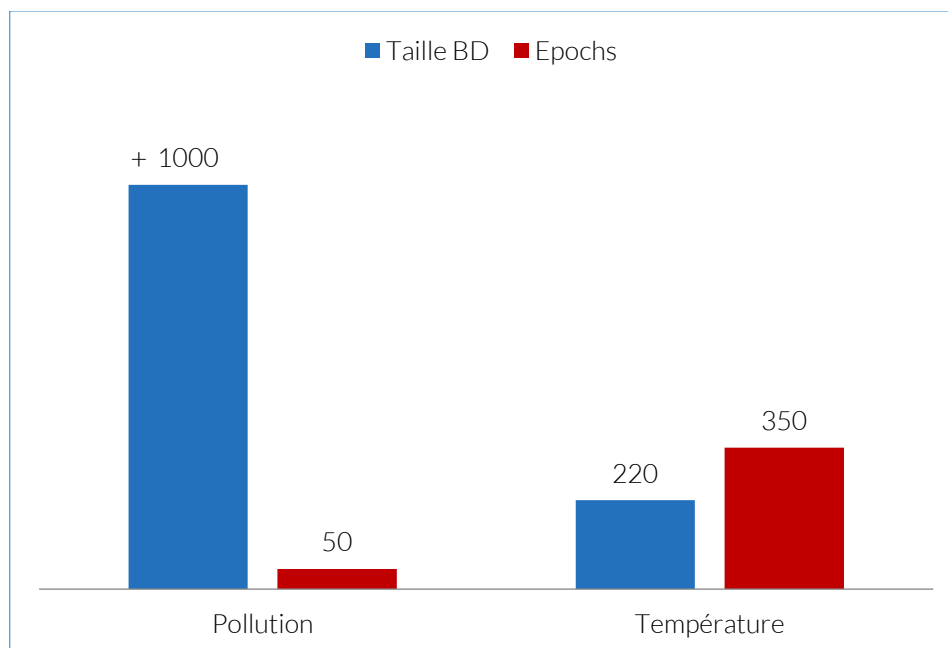


Figure 43 - Comparaison de la taille de la base de données et le nombre d'epochs

## 2. INTERFACE WEB DE LA RASPBERRY PI

Tous les résultats obtenus sont affichés dans une interface Web au sein de la Raspberry pi et sont actualisés automatiquement à chaque nouvelle prédiction.

Ci-dessous quelque capture de l'interface :

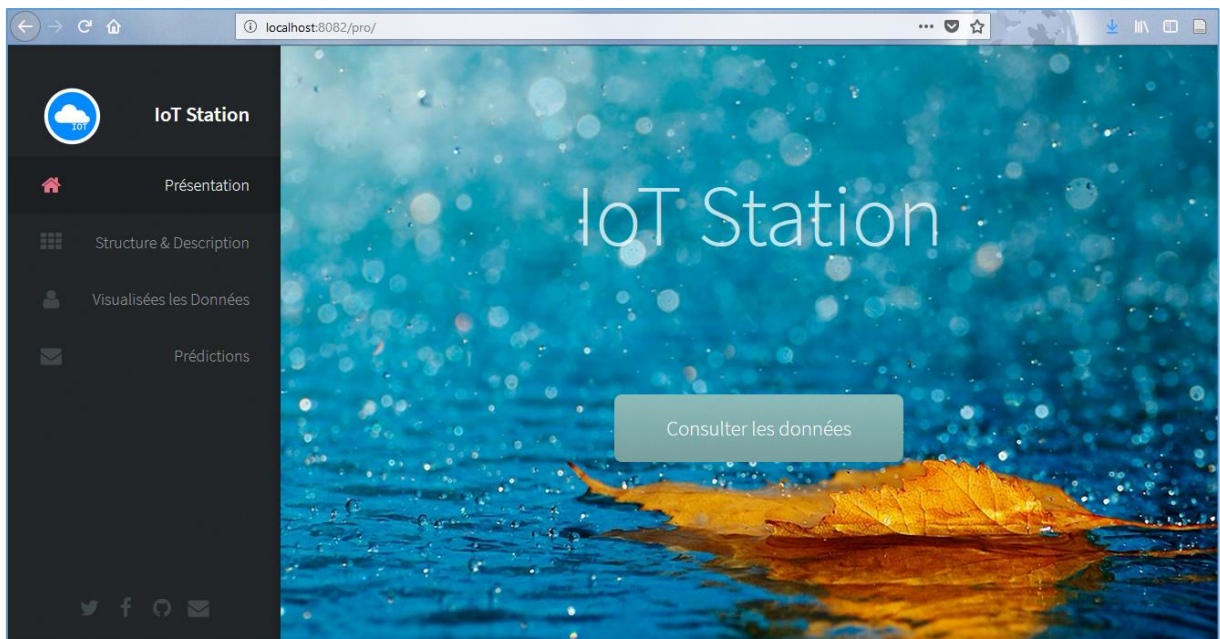


Figure 44 - Interface web, Présentation

Cette interface nous permet de :

- Voir la structure du système LoRa d'où viennent les données météorologiques
- Visualisées les enregistrements récentes de la base de données, ainsi que tous les valeurs de la température dans une courbe.
- Et finalement, voir les résultats de l'entrainement et de la prédiction des données.

La capture suivante montre la Structure du Réseau :

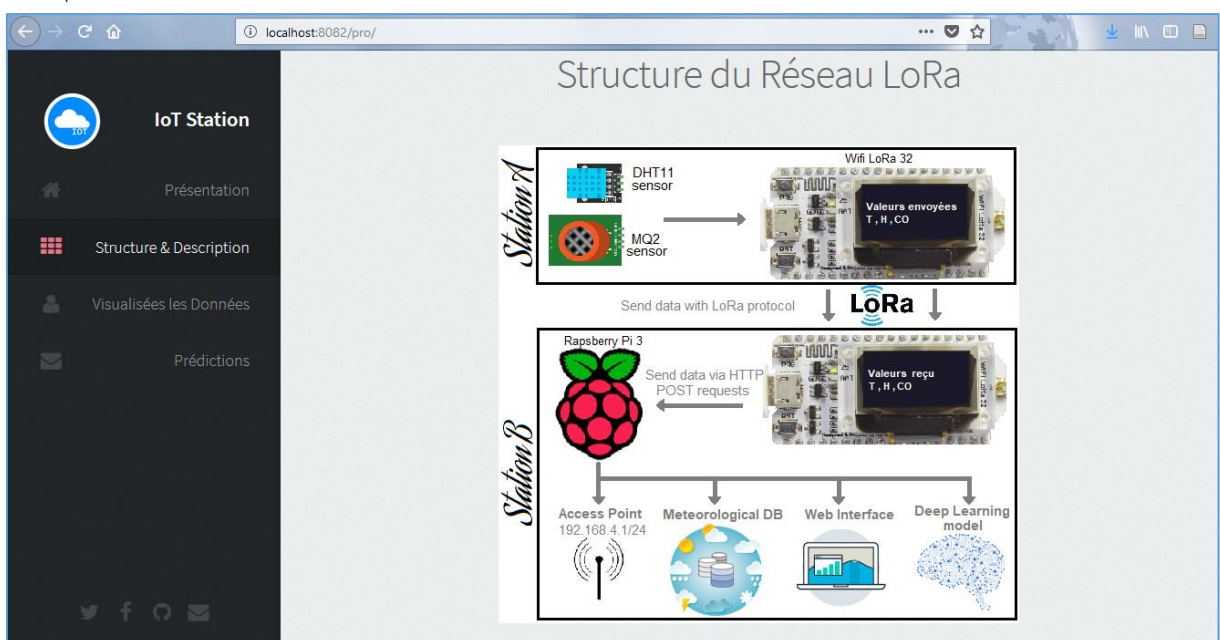


Figure 45 - Interface web, Structure

La visualisation des données :

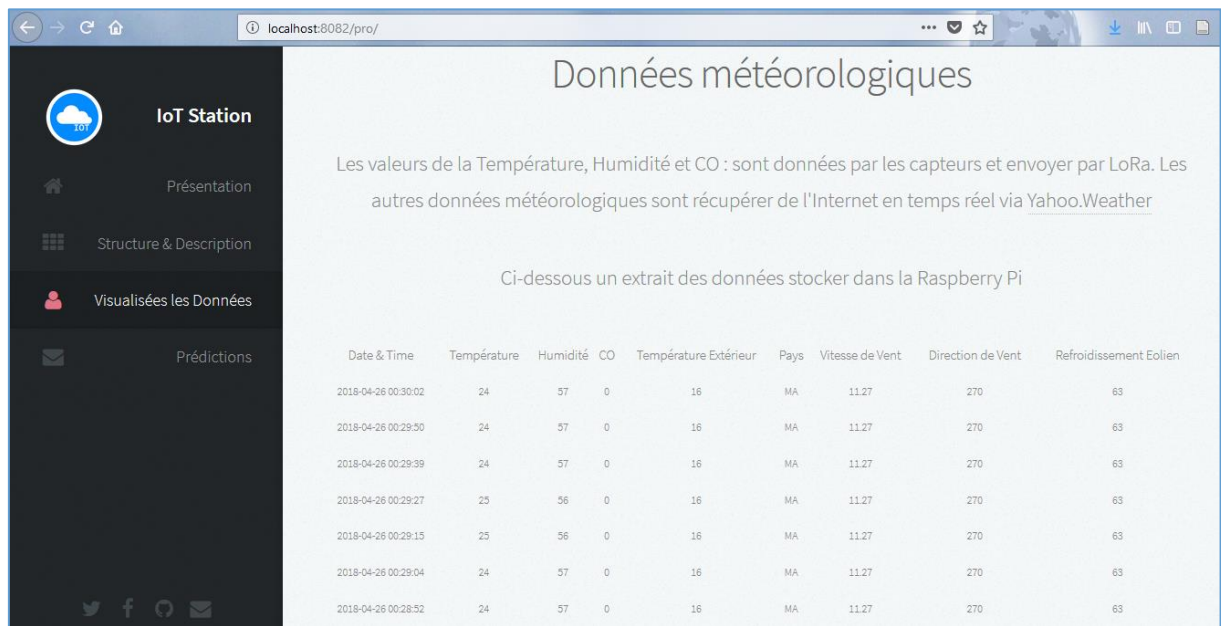


Figure 46 - Interface web, visualisation des données (1)

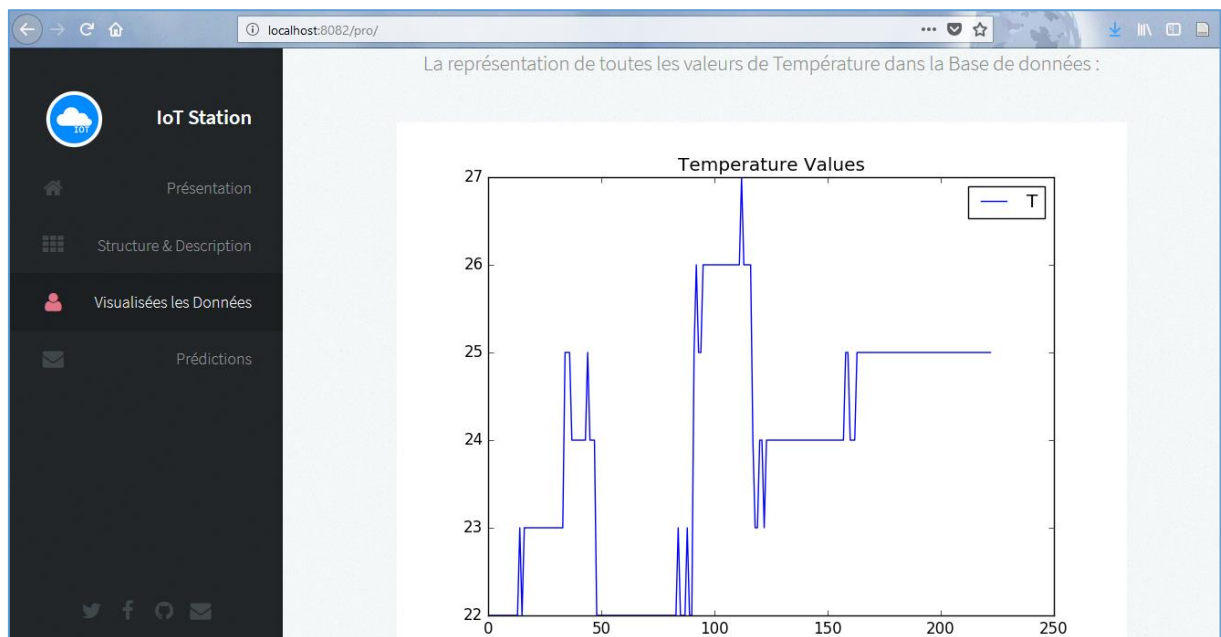


Figure 47 - Interface web, visualisation des données (2)

Prédictions et résultats :

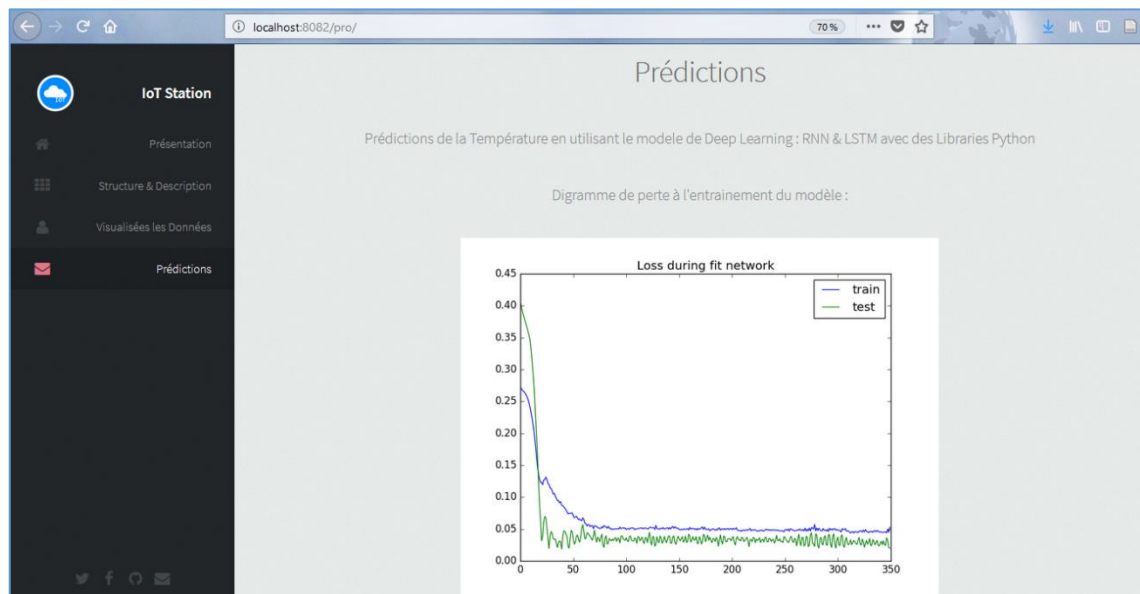


Figure 48 - Interface web, Prédiction (1)

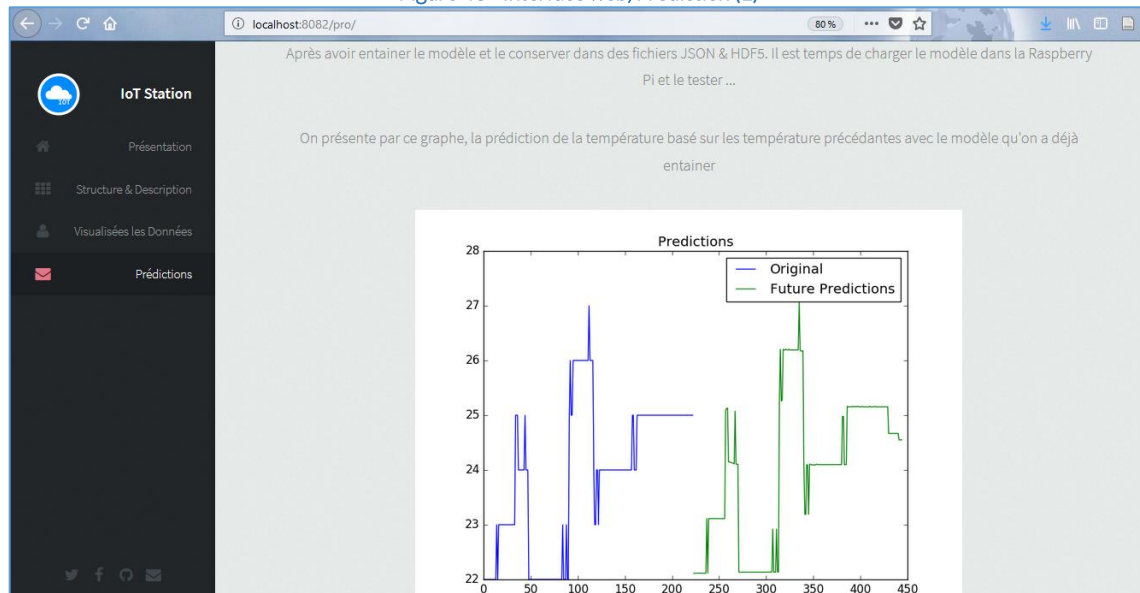


Figure 49 - Interface web, Prédiction (2)



Figure 50 - Interface web, Prédiction (3)

# CONCLUSION GENERALE

La richesse de ce projet réside dans la combinaison de l'aspect purement IoT et les réseaux LoRaWAN avec l'aspect de l'intelligence artificielle. Dans l'implémentation des réseaux de neurones récurrents pour la prédiction de la température d'après divers données météorologiques.

Les objectifs ont été remplis puisqu'un état de l'art sur l'IoT, le protocole LoRa et son protocole associé LoRaWAN, ainsi que l'analyse des données par l'apprentissage profond a été réalisé. Ainsi que la réalisation de la station IoT de test qui permet de visualiser les données en temps réel notamment les prédictions réalisées.

Cependant, comme ces domaines subissent régulièrement des améliorations, il faut garder à l'esprit que les informations rassemblées dans ce manuscrit peuvent devenir désuètes.

Néanmoins, l'ouverture des réseaux de capteurs à Internet présente un problème sérieux du point de vue sécuritaire car les réseaux de capteurs connectés à Internet deviennent accessibles à distance et d'une façon ubiquitaire par n'importe quel hôte malicieux sur Internet. Les contraintes des réseaux de capteurs, ainsi que l'hétérogénéité qui caractérise les communications entre les hôtes réguliers et les nœuds capteurs dans l'Internet des objets sont les vulnérabilités majeures de l'intégration qui est à l'origine de diverses attaques, essentiellement les attaques par déni de service (DoS).

En effet, les solutions de sécurité déjà approuvées et qui ont été proposées pour l'Internet classique, ne peuvent pas être directement projetées sur les nœuds capteurs contraints. Dans ce contexte, plusieurs solutions ont récemment intensivement investigué le problème de la sécurité des réseaux de capteurs connectés à Internet [13]. Ces solutions ne sont pas bien adaptées car elles ne prennent pas en considération la particularité des réseaux de capteurs dans l'IoT en tant que réseaux particuliers et négligent parfois même la contrainte énergétique sévère des nœuds capteurs.

Comme perspective, l'objectif sera orienté vers la sécurité de ces systèmes.



# REFÉRENCES

- [1] A. Augustin, J. Yi, T. Clausen, and W. M. Townsley, 'A Study of LoRa: Long Range & Low Power Networks for the Internet of Things', *Sensors*, vol. 16, no. 9, p. 1466, Sep. 2016.
- [2] W. Guibene, K. E. Nolan, and M. Y. Kelly, 'Survey on Clean Slate Cellular-IoT Standard Proposals', in *2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing*, 2015, pp. 1596–1599.
- [3] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, 'Internet of Things for Smart Cities', *IEEE Internet Things J.*, vol. 1, no. 1, pp. 22–32, Feb. 2014.
- [4] somia sahraoui, 'Mécanismes de sécurité pour l'intégration des RCSFs dans l'IoT (Internet of Things)', Thèse Doctorale, Université Mustapha Ben Boulaid Batna 2, Algérie, 2016.
- [5] M. Weiser, 'The Computer for the 21st Century', *SIGMOBILE Mob Comput Commun Rev*, vol. 3, no. 3, pp. 3–11, Jul. 1999.
- [6] S. Mann, 'Smart clothing: The wearable computer and wearcam', *Pers. Technol.*, vol. 1, no. 1, pp. 21–27, Mar. 1997.
- [7] K. Ashton, 'That "Internet of Things" Thing', *RFID J.*, 2011.
- [8] J.-P. Vasseur and A. Dunkels, *Interconnecting Smart Objects with IP: The Next Internet*. Morgan Kaufmann, 2010.
- [9] T. Perumal, S. K. Datta, and C. Bonnet, 'IoT device management framework for smart home scenarios', in *2015 IEEE 4th Global Conference on Consumer Electronics (GCCE)*, 2015, pp. 54–55.
- [10] K. Mekki, E. Bajic, F. Chaxel, and F. Meyer, 'A comparative study of LPWAN technologies for large-scale IoT deployment', *ICT Express*, Jan. 2018.
- [11] L. Vangelista, A. Zanella, and M. Zorzi, 'Long-Range IoT Technologies: The Dawn of LoRa<sup>TM</sup>', in *Future Access Enablers for Ubiquitous and Intelligent Infrastructures*, 2015, pp. 51–58.
- [12] J. de C. Silva, J. J. P. C. Rodrigues, A. M. Alberti, P. Solic, and A. L. L. Aquino, 'LoRaWAN #x2014; A low power WAN protocol for Internet of Things: A review and opportunities', in *2017 2nd International Multidisciplinary Conference on Computer and Energy Science (SpliTech)*, 2017, pp. 1–6.
- [13] J. Yang and B. Fang, 'Security model and key technologies for the Internet of things', *J. China Univ. Posts Telecommun.*, vol. 18, pp. 109–112, Dec. 2011.
- [14] S. Babar, P. Mahalle, A. Stango, N. Prasad, and R. Prasad, 'Proposed Security Model and Threat Taxonomy for the Internet of Things (IoT)', in *Recent Trends in Network Security and Applications*, 2010, pp. 420–429.

- [15] Symantec, 'Rapport Internet Security Threat Report (ISTR) 2018 | Symantec Canada', 2018.
- [16] B. Jan *et al.*, 'Deep learning in big data Analytics: A comparative study', *Comput. Electr. Eng.*, Dec. 2017.
- [17] G. Gelly, 'Réseaux de neurones récurrents pour le traitement automatique de la parole', phdthesis, Université Paris-Saclay, 2017.
- [18] I. Sutskever, O. Vinyals, and Q. V. Le, 'Sequence to Sequence Learning with Neural Networks', in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2014, pp. 3104–3112.
- [19] S. Hochreiter and J. Schmidhuber, 'Long Short-Term Memory', *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.
- [20] Q. Zhang, H. Wang, J. Dong, G. Zhong, and X. Sun, 'Prediction of Sea Surface Temperature Using Long Short-Term Memory', *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 10, pp. 1745–1749, Oct. 2017.