



Université Hassan II de Casablanca

Faculté des sciences et techniques de Mohammed VI

Département d'Informatique

Projet de Fin d'Etudes

**LICENCE SCIENCES ET TECHNIQUES
INFORMATIQUE, RÉSEAUX ET MULTIMÉDIA**

**Sujet :
Chatbot Touristique**

Présenté par :

AITHOUNA Basma

FAKIL Mohammed

BENBELLA Taha

Devant les membres de jury :

Pr. AYAD Habib

Encadrant (FSTM)

Pr. ADIB Abdellah

Rapporteur (FSTM)

Pr. ELBOUNY Lahcen

Examineur (FSTM)

Dédicace

Nous dédions ce projet de fin d'études.

A nos chères mères.

En témoignage de notre profonde gratitude et de notre incontestable reconnaissance, pour tous les sacrifices qu'elles nous contentent, toute la confiance qu'elles nous accordent et tout l'amour dont elles nous entourent.

A nos chers pères.

Qui sont les meilleurs pères dans ce monde, grâce à leurs encouragements, leur confiance et leur soutien moral et matériel et pour leur amour infini, en exprimant nos gratitude, notre profond amour et notre passion.

A nos chers frères et sœurs En leurs espérant le plein succès dans leur vie. A toutes nos familles, tous les étudiants de la Faculté des Sciences et Techniques Mohammedia et nos amis et tous ceux qui nous sont chers.

A nos chers enseignants.

Pour leurs efforts qui nous ont permis d'acquérir une formation de qualité, tous ceux qui ont collaboré de près ou de loin à l'élaboration de ce travail.

Remerciement

Avant d'ouvrir ce rapport, nous profitons de l'occasion pour remercier **ALLAH** pour la volonté, la force et la santé qu'il nous a donné afin de réaliser ce travail.

Nous tenons à remercier toute l'équipe pédagogique de la Licence Informatique, Réseaux et Multimédias.

En personne du chef de filière, Monsieur **MOUMKINE Noureddine** et nos enseignants qui nous ont accompagnés pendant cette année et qui ont partagé avec nous leurs expériences et qui nous ont encouragés sans cesse.

Nos remerciements sont également adressés à notre encadrant Monsieur **AYAD Habib**, nos encadrants à la FSTM pour leurs conseils et recommandations pertinentes durant toute la période de ce travail, ils nous ont fait le grand honneur d'accepter de nous diriger dans ce travail avec bienveillance et rigueur.

Nos remerciements vont aussi à l'encontre de nos chers parents qui n'ont ménagés aucun effort et qui ne cesseront de le faire pour nous soutenir à tout moment de notre vie.

Finalement, nous remercions l'ensemble des membres du jury qui nous ont fait l'honneur d'accepter d'évaluer et de juger notre travail.

Résumé

Dans le cadre de notre formation à la faculté des sciences et techniques Mohammedia, on a eu l'opportunité de réaliser un stage d'un mois dans l'entreprise de notre choix. Mais à cause de la pandémie que le monde fait face, on n'a pas pu faire un stage externe dans une entreprise, c'est pour cela que nos enseignants et toute l'équipe pédagogique de la Licence Informatique, Réseaux et Multimédias nous ont demandé des projets en nous encadrant en présentiel.

L'objectif de ce travail est la conception et la réalisation d'une application Web de conversation touristique (chatbot) qui permettra à l'utilisateur d'avoir une idée générale sur les différents endroits (hôtels, restaurants, monument historiques) dans la ville Marrakech.

La lecture des rapports scientifiques, des ouvrages de référence et de la documentation technique des outils disponibles, nous ont permis d'acquérir des connaissances permettant de comprendre les procédures méthodiques et de les appliquer correctement. Cela comprenait la collecte et l'enregistrement des données de formation pour les modèles, et le développement et la formation du prototype.

Chers lecteurs, nous optons que ce modeste rapport vous fournira autant d'informations bénéfiques que possible sur notre projet.

Table des matières

DEDICACE	2
REMERCIEMENT	3
RESUME	4
INTRODUCTION.....	8
1 CHAPITRE.....	9
1.1 ORGANISME D'ACCUEIL :	10
1.1.1 PRESENTATION DE LA FACULTE :	10
1.1.2 SES MISSION :	10
1.1.3 SES VALEURS :	11
1.1.4 ORGANISME D'ACCUEIL :	11
1.1.5 DEPARTEMENT INFORMATIQUE DE LA FSTM :	12
1.2 CONTEXTE GENERALE ET OBJECTIF :	13
1.2.1 ETUDE DE L'EXISTANT :	13
1.2.2 BESOINS FONCTIONNELS :	13
1.2.3 BESOINS TECHNIQUES :	14
1.3 CONCEPTION DU PROJET :	14
1.3.1 DIAGRAMME DE CAS D'UTILISATIONS :	15
1.3.2 DIAGRAMME DE PERT :	15
1.3.3 DIAGRAMMES DE GANTT :	16
2 CHAPITRE.....	17
2.1 INTRODUCTION :	18
2.2 CHATBOT :	18
2.2.1 DEFINITION :	18
2.2.2 DEVELOPPEMENT ET HISTOIRE DES CHATBOTS :	19
2.2.3 TYPES ET ARCHITECTURES DES CHATBOTS :	21
2.2.3.1 Systèmes basés sur des règles :	22
2.2.3.2 Systèmes intelligents :	23
2.3 CONCLUSION :	24
3 CHAPITRE.....	25
3.1 INTRODUCTION :	26
3.2 RESEAUX DE NEURONES :	26
3.2.1 RESEAUX NEURONES ARTIFICIELS :	27
3.2.2 FONCTIONNEMENT D'UN RESEAU NEURONAL ARTIFICIEL.....	28

3.2.3	FONCTION D'ACTIVATION :	29
3.2.4	APPRENTISSAGE DES RESEAUX NEURONES :	32
3.2.4.1	L'algorithme de la retro-propagation du gradient d'erreur :	32
3.2.4.2	Fonction de perte (Loss function) :	33
3.2.4.3	Optimisation par le gradient descend :	34
3.2.5	LES TYPES DE RESEAU DE NEURONES :	35
3.3	TRAITEMENT NUMERIQUE DU LANGAGE :	38
3.3.1	DEFINITION :	38
3.3.2	NLP ET CHATBOTS :	38
3.3.3	PRINCIPAUX CHALLENGES :	38
3.3.4	ETAPE DE BOW :	39
3.3.5	LIMITE DE BOW :	42
3.4	CONCLUSION :	42
4	CHAPITRE.....	43
4.1	PYTHON :	44
4.2	FLASK :	45
4.2.1	FRAMEWORK :	45
4.2.2	FLASK :	46
4.3	HTML & CSS :	47
4.3.1	HTML :	47
4.3.2	CSS :	47
4.4	JAVASCRIPT :	48
4.5	CONCLUSION :	48
5	CHAPITRE.....	49
5.1	BASE DE DONNEES :	50
5.1.1	JSON :	50
5.1.2	LA STRUCTURE DE LA BASE DE DONNEES :	50
5.1.3	L'AUGMENTATION DES DONNEES :	51
5.1.4	CHATITO :	52
5.2	L'APPLICATION WEB :	53
	CONCLUSION GENERALE :	55
	REFERENCES	56

Table de figures

Figure 1:Organisation générale	11
Figure 2 : Diagramme de cas d'utilisations.....	15
Figure 3 Diagramme de Pert	15
Figure 4 : Plannification du projet	16
Figure 5 : Le test de Turing	19
Figure 6 : Messenger Platform	20
Figure 7 : Chronologie de l'histoire des chatbots	21
Figure 8 : Classification des Chatbots	21
Figure 9 : Approche basée sur les règles.....	22
Figure 10 : Architecture retrieval.....	23
Figure 11 : Architecture générative	24
Figure 12 : Neurone biologique	26
Figure 13 : Réseau de Neurone Artificiel	27
Figure 14 : Correspondance entre neurone artificiel et neurone biologique	28
Figure 15 : Géométrie du neurone artificiel.....	29
Figure 16 : Fonction ReLU-Rectified Linear Unit.....	30
Figure 17 : Fonction Sigmoid	31
Figure 18 : Fonction Softmax.....	31
Figure 19 : Apprentissage par rétro propagation	33
Figure 20 : Modélisation de la fonction de perte	33
Figure 21 : Minimum local de la fonction de perte	34
Figure 22 : Perception simple.....	36
Figure 23 : Perception Multicouche.....	36
Figure 24 : Réseau neuronal convolutif	37
Figure 25 : Réseau neuronal récurrent	37
Figure 26 : Etapes de BOW	39
Figure 27 Fonctionnement d'un EDA	51
Figure 28 Interface de Chatito.....	52
Figure 29 Interface d'accueil	53
Figure 30 Interfaces de chat	54

Table des tableaux

Tableau 1 Présentation des enseignants	12
Tableau 2 Fonction d'activation Vs Modèle	32

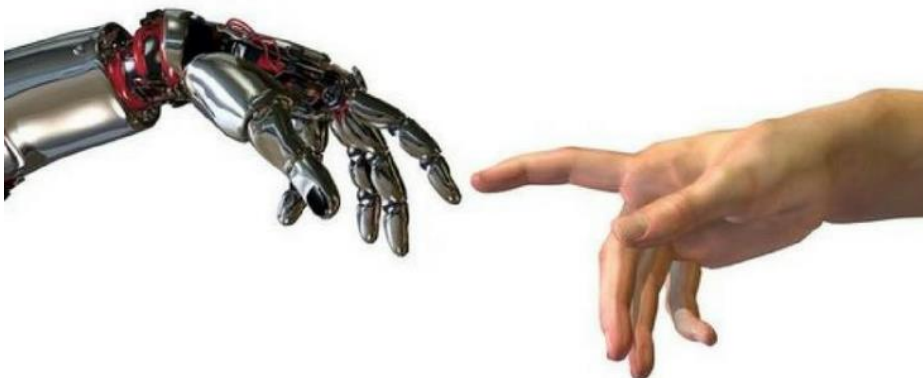
Introduction

La manifestation la plus visible de l'intelligence artificielle (IA) est, sans aucun doute, les « **chatbot** » ou « **agents conversationnels** ». Ce sont des assistants virtuels capables de simuler une conversation en imitant le comportement humain, que ce soit à l'écrit ou à l'oral. Grâce à la technologie moderne, les entreprises de voyages peuvent désormais communiquer avec des clients grâce à des robots utilisant des applications de chat.

Il existe 2 types d'interfaces : les chatbot textes comme par exemple Messenger, et les interfaces voix qui permettent d'interagir de façon naturelle avec une machine. Depuis l'ouverture des chatbot sur Messenger en 2016, plus de 100 000 chatbot ont été créés. Le chatbot est un outil qui est en train de révolutionner la relation client, il est devenu nécessaire dans certains secteurs et le deviendra dans d'autres.

Dans ce projet nous cherchons à développer un ChatBot touristique qui permet aux clients de poser des questions à propos le tourisme à Marrakech et qui gère la conversation d'une manière naturelle.

L'IA assure cette tâche grâce à un ensemble de techniques (Deep Learning, Machine Learning, Traitement du Langage Humain) permettant d'imiter le comportement humain et d'avoir une intelligence indistinctement de celle de l'être humain dans une conversation.



Chapitre 1

Présentation du cadre du projet

Dans ce premier chapitre nous allons présenter le cadre où ce travail a été effectués. Il est divisé en 3 parties : l'organisme d'accueil, le contexte et l'objectif et la partie conception.

1.1 Organisme d'accueil :

1.1.1 Présentation de la faculté :

La Faculté de Sciences et Techniques de Mohammedia (FSTM), composante de l'Université Hassan II de Casablanca, fait partie d'un réseau de huit FST à travers le Maroc dont la vocation est la formation universitaire dans les sciences et techniques, la FSTM, implantée dans une zone à forte activité industrielle, a toujours veillé à s'intégrer dans son environnement en disposant des cursus aboutissant à des profils qui répondent par excellence aux besoins socio-économiques. Elle œuvre depuis son inauguration, en 1994, à offrir une formation technique et en ingénierie de qualité, et s'est distinguée par la pertinence de sa recherche. La FST de Mohammedia compte 186 enseignants chercheurs, 57 personnels administratifs et techniques et plus de 3000 étudiants dont 750 en dernière année de licence 260 en master et 440 en filières d'ingénieurs. Elle délivre plus que 550 diplômes par an. C'est un établissement à accès régulé où l'entrée se fait après une sélection basée sur les notes au baccalauréat de matières définies en fonction du parcours choisi (MIP ou BCG). La FSTM compte 8 départements et 13 laboratoires. Les formations de la recherche au sein de la FSTM couvrent différents champs disciplinaires en Biologie, Mathématiques et Physiques. La FSTM propose une carte de formation riche et relativement complète en Licence, Master Sciences et Techniques et Filières d'ingénieurs.

1.1.2 Ses Mission :

La Faculté des Sciences et Techniques de Mohammedia (FSTM) regroupe des structures qui s'investissent pleinement dans les domaines de la formation et de la recherche scientifique pour contribuer au développement du Maroc. Ces structures sont engagées pour :

— Assurer des formations scientifiques et techniques de qualité en adéquation avec le marché de l'emploi.

- Développer la recherche scientifique fondamentale et appliquée pour contribuer à la production du savoir, répondre aux besoins des entreprises et encourager l'innovation.
- Dispenser des formations continues pour répondre aux besoins du secteur socio-économique.

1.1.3 Ses Valeurs :

La FSTM transmet des valeurs fondées sur l'intégrité intellectuelle et morale :

- La FSTM s'engage à développer le professionnalisme, les sens des responsabilités, la créativité et le respect des autres.
- La FSTM s'engage à respecter les principes d'équité, d'égalité des chances et de multiculturalisme.
- La FSTM s'engage à émuler l'excellence, la créativité et l'innovation.

1.1.4 Organisme d'accueil :

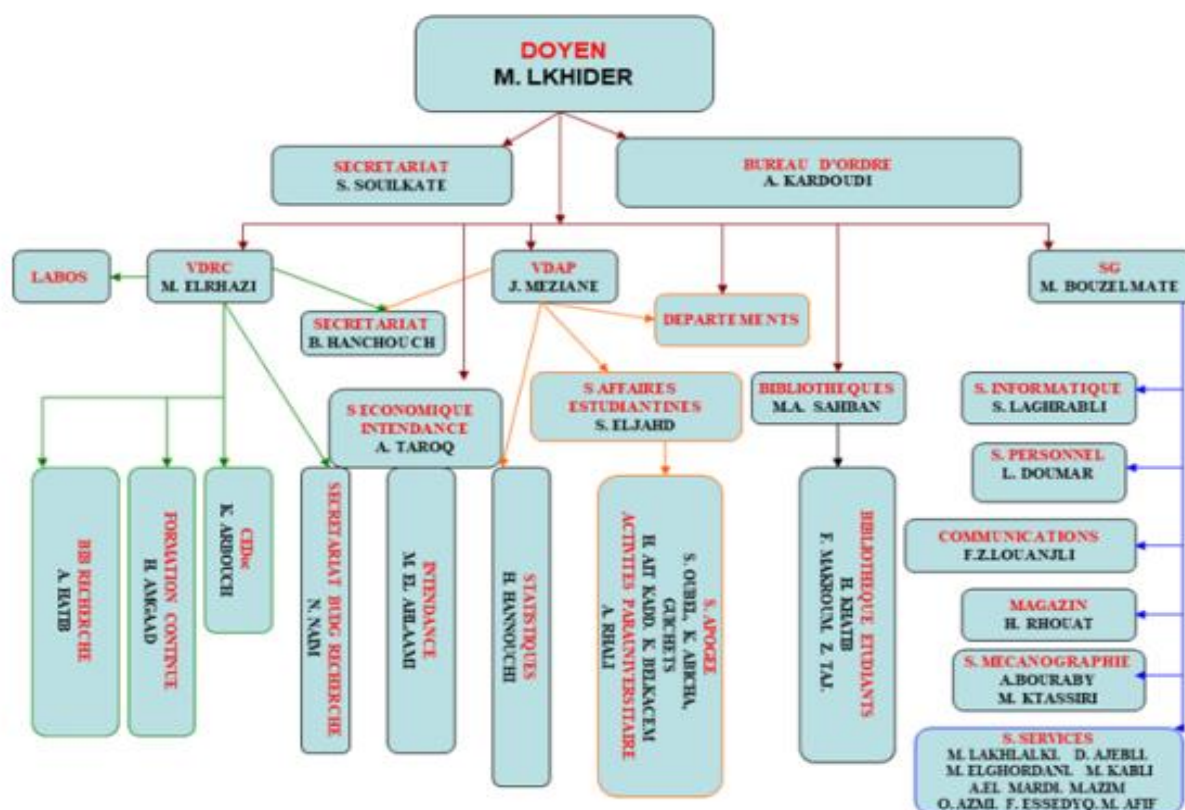


Figure 1: Organisation générale

1.1.5 Département Informatique de la FSTM :

Le département Informatique a pour mission globale de mettre tout en œuvre, dans le cadre de l'Université Hassan II de Casablanca, pour favoriser la formation et l'émergence du génie informatique et du génie logiciel. Par son implication dans des programmes à tous les cycles, le département vise à donner une formation universitaire de qualité qui répond, de façon dynamique et pratique, aux besoins de la société. Il contribue ainsi à une large diffusion des Sciences et Technologies de l'Information et de la Communication (STIC) dans différents secteurs d'activités (industrie, services, transport, administration, enseignement...).

Le département s'est structuré et renforcé sur le plan de la recherche, en mettant en place une organisation s'appuyant sur quatre Unités de Recherche chapeauté par un laboratoire : le Laboratoire Informatique de Mohammedia (LIM).

Par le rayonnement des professeurs du LIM, autant par leurs activités innovatrices que par la publication de leurs résultats de recherche, le LIM contribue à l'actualisation, l'avancement, la création et la diffusion d'idées nouvelles reliées à l'ingénierie et à l'application de l'informatique.

Nom	Prénom	Grade
ADIB	Abdellah	Professeur de l'Enseignement Supérieur
ANTER	Samir	Professeur Assistant
BEKKHOUCHA	Abdelkrim	Professeur de l'Enseignement Supérieur
BOUAOUA	Naima	Professeur Habilité
BOULMAKOUL	Azedine	Professeur de l'Enseignement Supérieur
CHANTIT	Safima	Professeur Assistant
DOUZI	Khadija	Professeur de l'Enseignement Supérieur
EL BEGGAR	Omar	Professeur Assistant
EL BOUZIRI	Adil	Professeur Assistant
ELIABRI	Amina	Professeur Assistant
HSSINA	Badr	Professeur Assistant
KHALIL	Mohammed	Professeur Assistant
KHOUKHI	Faddoul	Professeur Habilité
KISSI	Mohamed	Professeur de l'Enseignement Supérieur
LEGHRIS	Cherkaoui	Professeur Habilité
MOUMKINE	Nouredine	Professeur Assistant
RAMDANI	Mohammed	Professeur de l'Enseignement Supérieur
SADDOUNE	Mohammed	Professeur Habilité

Tableau 1 Présentation des enseignants

1.2 Contexte générale et Objectif :

Le but de ce stage de projet de fin d'étude est de concevoir et de réaliser une application web mobile à base d'un chatbot en anglais qui permet à l'utilisateur d'avoir une vue globale sur les différents hôtels, restaurants, attractions touristiques à Marrakech.

1.2.1 Etude de l'existant :

Pour notre projet de fin d'études, au niveau du Maroc il n'existe pas encore un chatbot touristique pour la ville de Marrakech. Par contre, Tanger a un chatbot qui s'agit d'une application informatique basée sur de l'intelligence artificielle et qui permettra aux futurs visiteurs de la ville de mieux organiser leurs déplacements en apportant des conseils aux touristes, entre autres, le projet est lancé par la CRT de Tanger-Tétouan-Al Hoceima en partenariat avec l'office national marocain du Tourisme.

1.2.2 Besoins fonctionnels :

L'objectif de l'Analyse fonctionnelle du besoin est de préciser les fonctions de service attendues à partir des prestations définies. Cette étape est primordiale pour une bonne réalisation du projet. L'application à réaliser doit répondre aux besoins fonctionnels suivants :

- La consultation, l'ajout et la modification des réponses proposées par le bot
- Affichage des informations par thèmes (hôtellerie, restaurations, attractions touristique)
- Détermination avec précision les réponses souhaitées
- Optimisation de la durée de répondre aux questions demandées par l'utilisateur

1.2.3 Besoins techniques :

- **Performance**

- L'application doit offrir aux utilisateurs des réponses fiables et correctes
- L'application doit être présentable
- L'application doit assurer l'intégrité et la cohérence de données (pas de conflits).

- **Ergonomie**

- L'application doit offrir une interface conviviale qui facilite l'utilisation et réduit le temps d'apprentissage du fonctionnement.
- L'utilisateur de l'application doit accéder à l'information rapidement : le temps de recherche et de réponse doit être minimal.

- **Confidentialité**

- La manipulation des questions, et l'envoi des réponses devraient passer en toute sécurité.

1.3 Conception du projet :

Cette partie consiste à identifier les diagrammes de cas d'utilisation de Pert et de Gantt réalisés pour mettre en œuvre le chatbot proposé. La motivation fondamentale de la modélisation est de fournir une démarche antérieure afin de réduire la complexité du système étudié lors de la conception et d'organiser la réalisation du projet en définissant des étapes de réalisation. Nous adoptons dans notre travail une approche objet basé sur un outil de modélisation UML.

En fait, UML (Unified Modeling Language) est un standard ouvert contrôlé par l'OMG, un consortium d'entreprises qui a été fondé pour construire des standards qui facilitent l'interopérabilité et plus spécifiquement, l'interopérabilité des systèmes orientés objet. UML est issu de l'unification de nombreux langages de modélisation graphique orientée objet. Il unifie à la fois les notations et les concepts orientés objets.

Pour modéliser ce projet, nous allons identifier trois diagrammes.

1.3.1 Diagramme de cas d'utilisations :

C'est un diagramme qui présente un intérêt pour l'analyse des besoins métier ce qui nous permettra de démarrer l'analyse orientée objet et identifier les classes candidates.

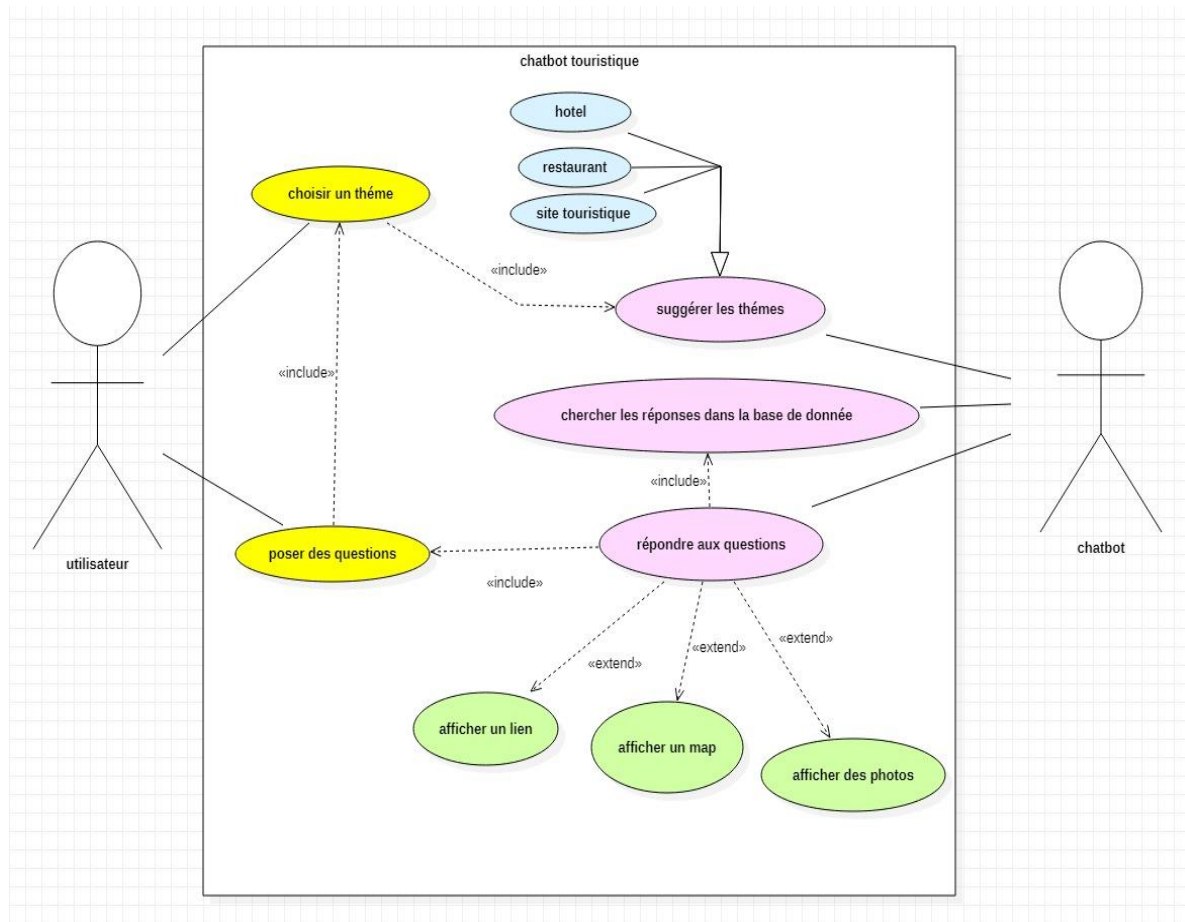


Figure 2 : Diagramme de cas d'utilisations

1.3.2 Diagramme de Pert :

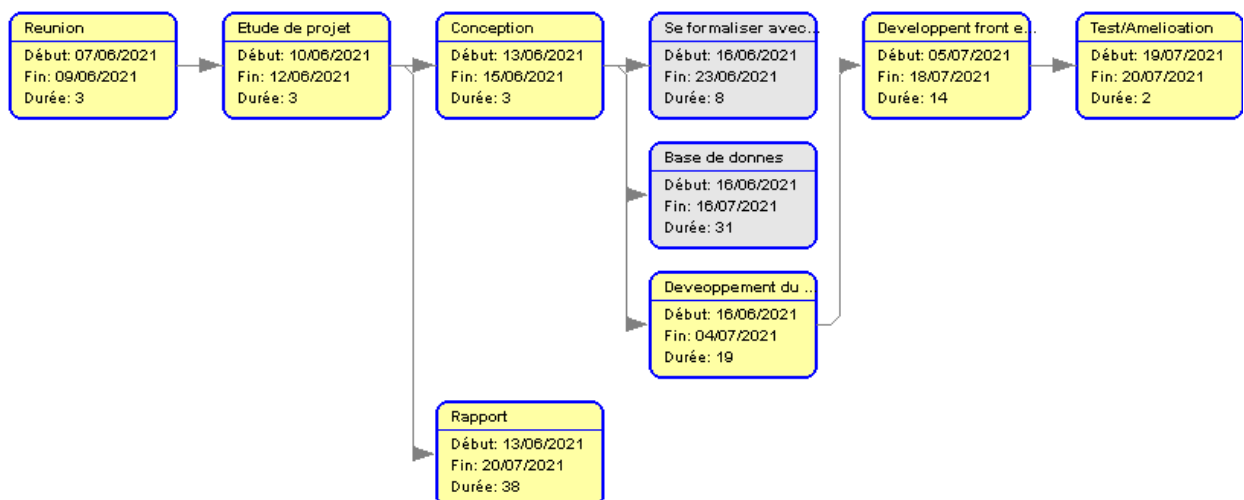


Figure 3 Diagramme de Pert

1.3.3 Diagrammes de Gantt :

C'est un outil utilisé en ordonnancement et en gestion de projet et permettant de visualiser dans le temps les diverses tâches composant un projet. Il s'agit d'une représentation d'un graphe connexe, valu et orienté, qui permet de représenter graphiquement l'avancement du projet.

GANTT project		
Nom	Date de début	Date de fin
• Reunion	07/06/2021	09/06/2021
• Etude de projet	10/06/2021	12/06/2021
• Conception	13/06/2021	15/06/2021
• Se formaliser avec le Machi...	16/06/2021	23/06/2021
• Base de donnes	16/06/2021	16/07/2021
• Deveoppement du code	16/06/2021	04/07/2021
• Developpent front end et in...	05/07/2021	18/07/2021
• Test/Amelioation	19/07/2021	20/07/2021
• Rapport	13/06/2021	20/07/2021

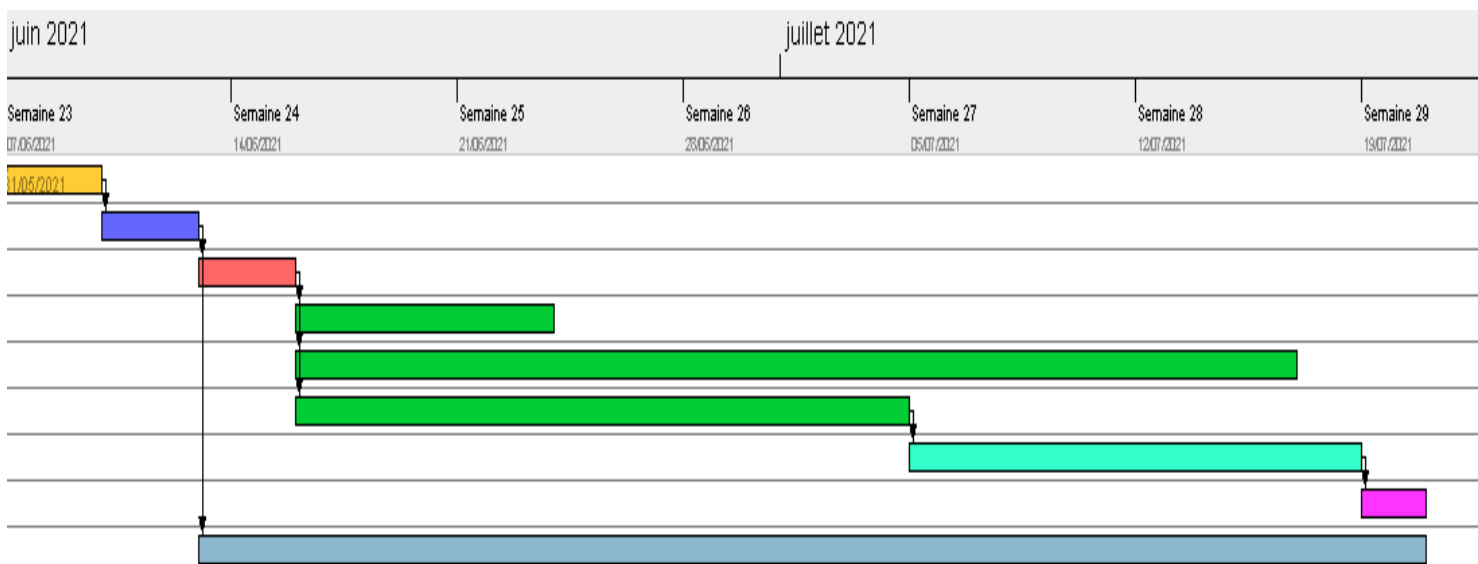


Figure 4 : Plannification du projet

Chapitre 2

Etat de l'art des Chatbots

2.1 Introduction :

Dès le moment où le futur voyageur désire faire des recherches dans le but de préparer ses prochaines vacances, il n'arrive pas forcément à le faire pendant les heures où un professionnel du tourisme ne sera pas disponible pour lui répondre à son besoin. C'est pour cela que la mise en place d'un chatbot touristique permettra aux voyageurs et aux touristes d'avoir toutes les informations désirées sur leur destination d'une manière permanente.

Donc dans ce chapitre, on va définir qu'est-ce qu'un Chatbot, savoir son histoire et ses différents types et son architecture.

2.2 Chatbot :

2.2.1 Définition :

D'après le dictionnaire d'*Oxford* :

« Un Chatbot est un programme informatique conçu pour simuler une conversation avec des utilisateurs humains, en particulier sur Internet » [1]

Donc un Chatbot, également connu sous 'Chatter bot' ou bot, ou aussi appelé agent conversationnel, est un « robot de discussion » qui communique par l'écrit en langage naturel.

CHATBOT = TO CHAT + ROBOT

Un chatbot peut prendre le format d'un agent virtuel sur un site en ligne ou bien désigner un assistant pour application mobiles, application de messagerie ou téléphone.

Les plus simples des chatbots sont des services basés sur des règles et des scénarios de dialogues prédéfinis.

Les plus puissants, ceux qui s'appuient sur l'intelligence artificielle, du Machine Learning et de la reconnaissance du langage naturel.

2.2.2 Développement et histoire des chatbots :

Les premiers agents conversationnels fonctionnaient grâce à une base de données avec un système simple de question/réponse. Par la suite, les chatbots ont évolué au rythme des avancées de l'intelligence artificielles (IA). [2]

Le premiers chatbot de l'histoire est « **ELIZA** » c'est un programme informatique inventé par Joseph Weizenbaum en 1966 qui simulait un psychiatre. Il fonctionnait par simple reconnaissance de forme et par reformulation des affirmations en question grâce à un système de mots-clés. Ce programme constituait déjà à l'époque de nombreuses limites mais il est ce qu'on pourrait appeler aujourd'hui le géniteur des chatbots actuels qui envahissent notre quotidien.



Ces innovations à l'époque étaient évaluée grâce au célèbre test de Turing, aussi appelé « The imitation Game » ou « Le jeu de l'imitation » qui a été imaginé par Alan Turing qui est considéré par tous les informaticiens comme le parrain de cette science. Dans son article « COMPUTING MACHINERY AND INTELLIGENCE », Turing a posé la question pour savoir est ce que les machines peuvent penser, il a reformulé cette question sous forme d'un jeu où la machine gagne s'elle peut faire ce que nous pouvons faire.

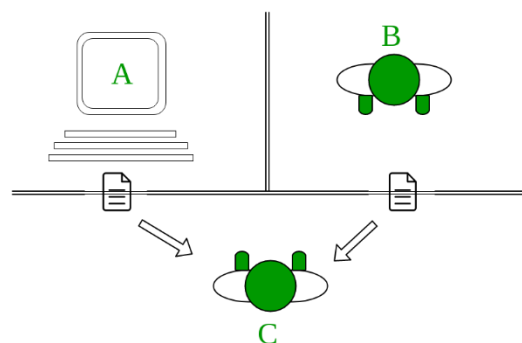


Figure 5 : Le test de Turing

En 1972, le scientifiques de Stanford « KENNETH COLBY » avait une autre idée, il a pu développer un programme qui simule comment un schizophrène paranoïde pense et il lui a donné le nom « **PARRY** » .Les gens le considéraient comme le premier patient de la thérapeute ELIZA.

Après le succès des deux premiers chatbots textuels, « **JABBERWACKY** » est devenu le premier chatbot à simuler la voix humaine. Le but de sa création était de permettre au chatbot de passer le test de Turing.

Sur les traces de JABBERWACKY, le chatbot nommé « **DR SBAITSO** » a été créé en 1992. Le bot avait une interface utilisateur où il imite les réponses d'un psychologue.

Dans les années 90's, Richard Wallace a développé un chatbot appelé « **ALICE** » (Artificial Linguistic Internet Computer Entity), il s'agit d'un programme de personnalité basé sur le langage de balisage de l'intelligence artificielle qui a remporté le prix Loebner dans la catégorie « l'ordinateur le plus humain » en 2000,2001 ,2004.

Au cours des années 2010, les géants du web Apple, Microsoft, Google et Amazon ont développé leurs propres assistants conversationnels tels que SIRI, CORTANA, GOOGLE, ALEXA ...Ces derniers ont introduit sur le terrain le concept de conversation ainsi que le dialogue axé sur les objectifs.

Un grand évènement majeur dans le domaine des chatbots a été la sortie de Messenger Platform par Facebook Messenger en 2016 ce qui a permis la création d'agents conversationnels.

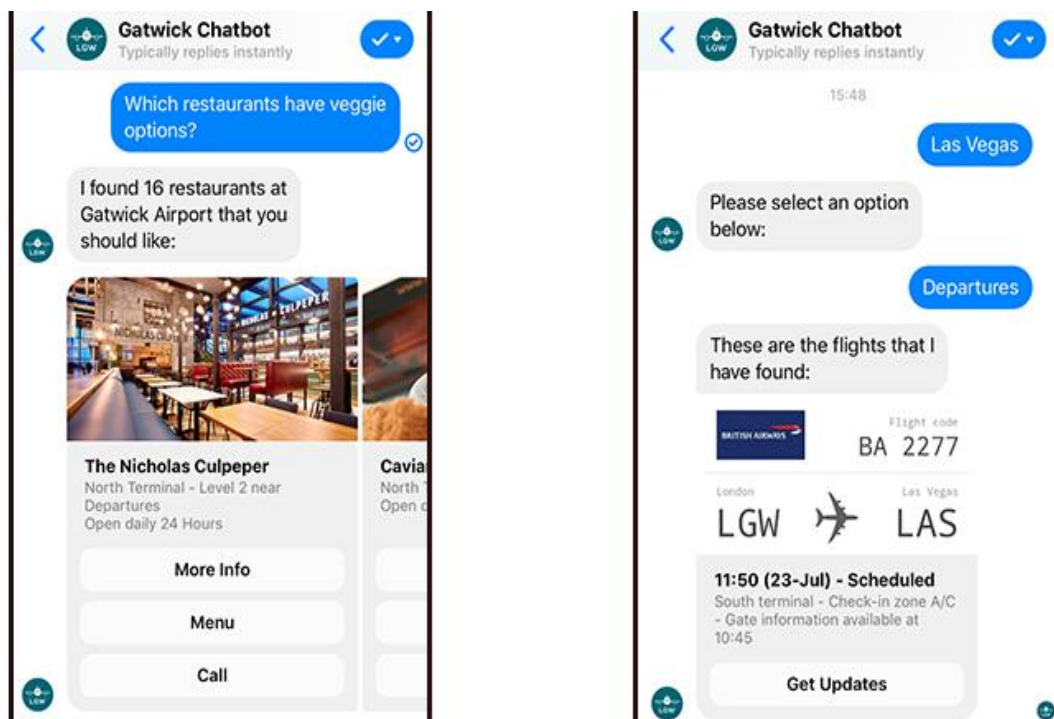


Figure 6 : Messenger Platform

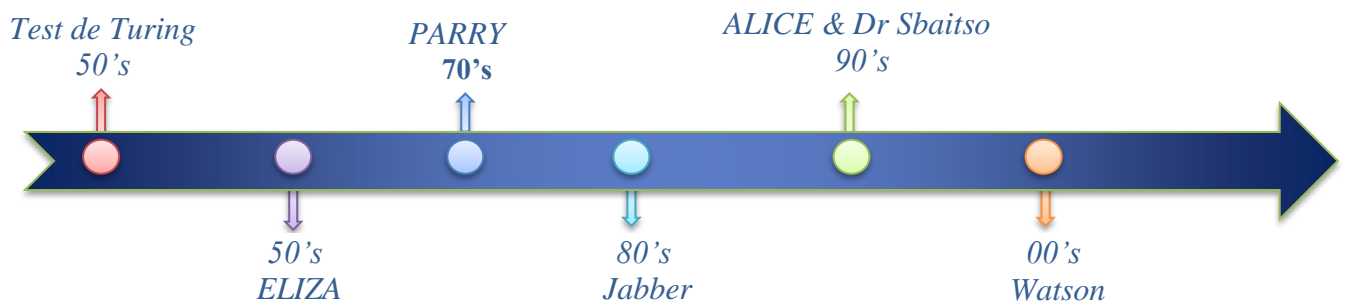


Figure 7 : Chronologie de l'histoire des chatbots

2.2.3 Types et architectures des chatbots :

Les chatbots sont généralement classés en fonction de leur capacité de conversation et de leur mécanisme de réponse, comme le montre le graphique ci-dessous :

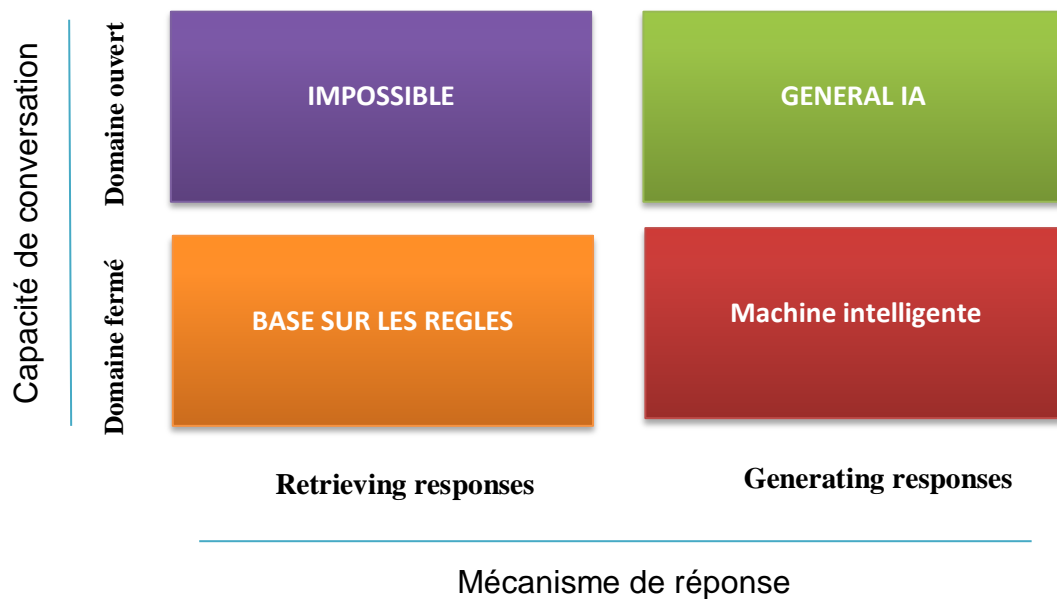


Figure 8 : Classification des Chatbots

- **Capacité de conversation** : Définit si le chatbot peut fournir des réponses significatives dans un domaine ouvert, ou il ne peut orienter les conversations que vers un domaine prédéfini (domaine fermé).

- **Mécanisme de réponse :** Fait références à la capacité du chatbot à récupérer les réponses (par exemple, à partir d'une base de données) ou créer dynamiquement de nouvelles réponses.

Pour mieux comprendre le fonctionnement des chatbots et choisir la meilleure option dans laquelle on va développer notre chatbot pour retirer un maximum de bénéfices, il faut savoir qu'il existe deux types de chatbots : les chatbots basés sur des règles et les chatbots basés sur l'intelligence artificielle (IA).

2.2.3.1 Systèmes basés sur des règles :

Les chatbots basés sur des règles sont assez simples. Ils disposent d'une base de données de réponses et d'un ensemble de règles qui les aident à trouver une réponse appropriée dans la base de données fournie. Ils ne peuvent pas générer leurs propres réponses, mais avec une vaste base de données de réponses et des règles intelligemment conçues, ils peuvent être très productifs et utiles.

Ces chatbots fonctionneront en recherchant des mots-clés spécifiques dans les entrées données par un utilisateur. Les mots-clés seront utilisés pour comprendre l'action que l'utilisateur veut entreprendre (l'intention de l'utilisateur). Une fois l'intention identifiée, le robot choisira une réponse appropriée à cette intention.

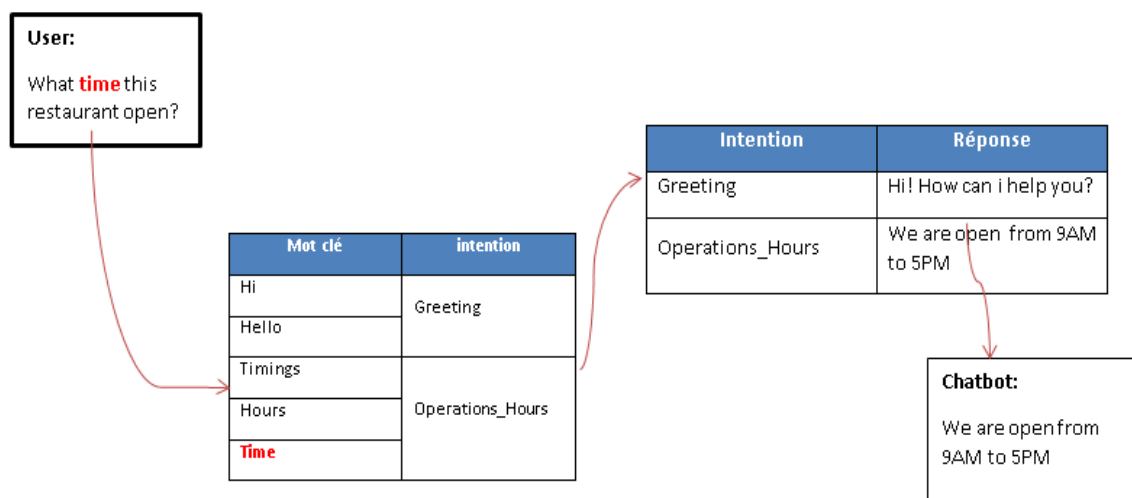
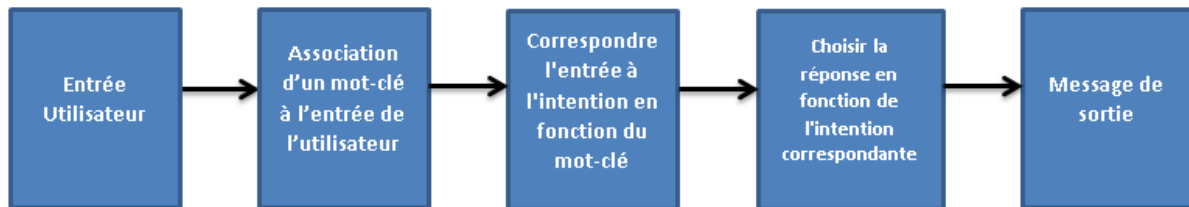


Figure 9 : Approche basée sur les règles

La liste des mots-clés que le robot recherchera et le dictionnaire des réponses seront construits manuellement en fonction du cas d'utilisation spécifique du chatbot.

Le Flux de la façon dont le chatbot traiterait les entrées est montré ci-dessous :



2.2.3.2 Systèmes intelligents :

Il existe de nombreux type d'approche pour créer un chatbot, certaines sont des approches basées sur l'intelligence artificielle comme : [3]

- **Retrieval-based** : Les robots basés sur cette technique disposent d'un référentiel ou une large base de données qu'ils l'utilisent pour répondre aux questions d'un utilisateur. Dans cette approche, le chatbot est formé pour classer la meilleure réponse parmi un ensemble fini des réponses prédéfinies et saisies manuellement par un humain. L'avantage de cette approche est que chaque réponse peut être contrôlée, et ainsi éviter les erreurs et les réponses inappropriées. Cependant cette méthode à ses propres limites elle semble trop rigide et les réponses peuvent ne pas sembler "humaines".

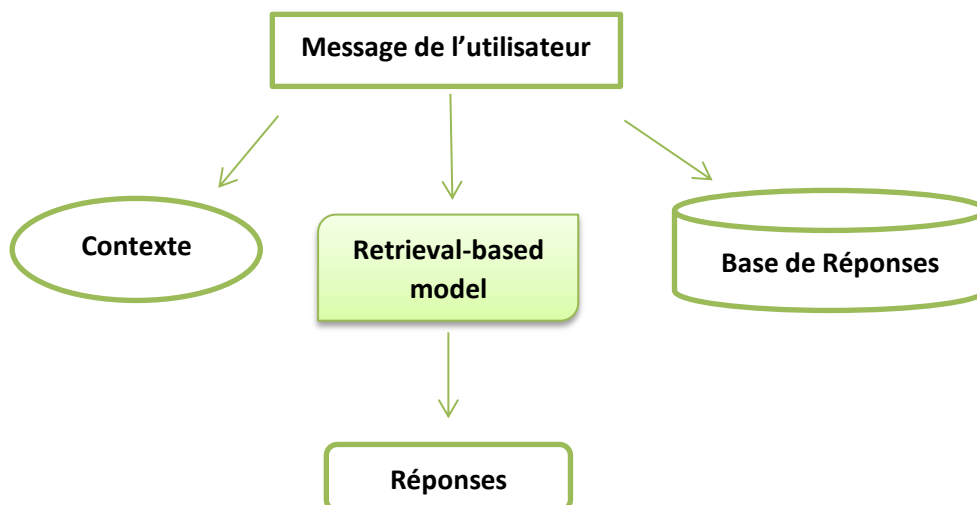


Figure 10 : Architecture retrieval

- **Generative model** : Une autre méthode de construction de chatbots consiste à utiliser un modèle génératif. Comme son nom l'indique, dans cette approche, le chatbot n'utilise aucune sorte de référentiel prédéfini, il peut générer une réponse par lui-même.

Le modèle génératif est utilisé pour le développement de robots intelligents qui sont assez avancés par nature car ils apprennent à partir de zéro. Au lieu de cela, ils sont formés en utilisant un grand nombre de conversations précédentes, sur la base desquelles les réponses sont générées.

La formation de ce type de robot nécessite d'investir beaucoup de temps et d'efforts. Cependant, vous ne pouvez toujours pas être sûr des réponses que le modèle va générer. Ce type de chatbot est très rarement utilisé, car il nécessite la mise en œuvre d'algorithmes complexes.

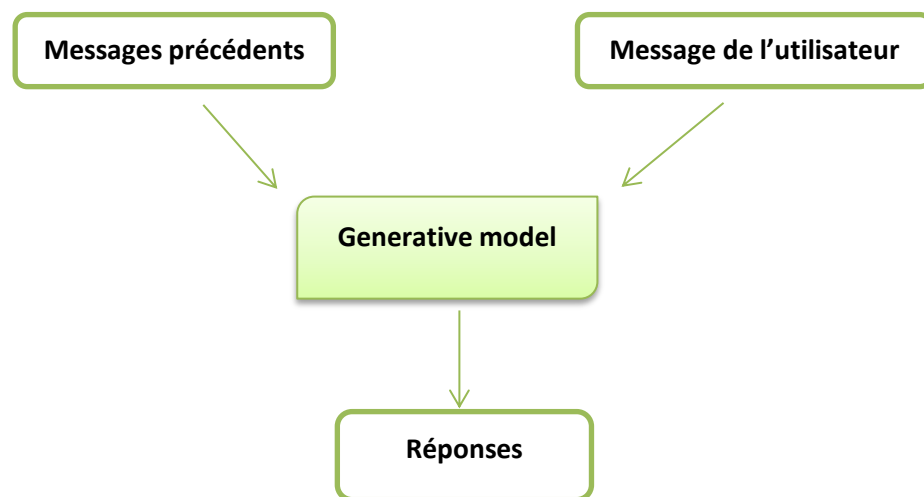


Figure 11 : Architecture générative

2.3 Conclusion :

Les chatbots connaissent une progression intéressante en parallèle avec celle de l'intelligence artificielle et ses différentes techniques.

Chapitre 3

L'apprentissage profond

3.1 Introduction :

L'interaction avec une machine via le langage naturel est rendue possible grâce aux progrès de l'intelligence artificielle. Le chatbot reconnaît le sens de votre phrase et vous fournit une réponse informative, maintient le contexte du dialogue d'une manière indifférenciable de l'humain. Pour y parvenir, l'intelligence artificielle d'un chatbot s'appuie sur **l'apprentissage profond** et aussi ce qu'on l'appelle le **Traitement du langage naturel** (NLP ou Natural Language Processing). Ce traitement du langage peut s'appliquer à un texte saisi dans une messagerie, mais également à la voix.

L'apprentissage profond ou le **Deep learning** est un ensemble de méthode d'apprentissage qui vise à modéliser des données avec des architectures complexes combinant différentes transformations non linéaires. Dans ce qui suit, nous aborderons les bases de cet apprentissage profond.

3.2 Réseaux de neurones :

Les réseaux de neurones, communément appelés des réseaux de neurones artificiels sont des imitations simples des fonctions d'un neurone dans le cerveau humain pour résoudre des problématiques d'apprentissage de la machine (Machine Learning).

Avant d'attaquer au fonctionnement précis des réseaux de neurones, on s'est dit qu'il serait intéressant de faire un parallèle avec les neurones biologiques. [4]

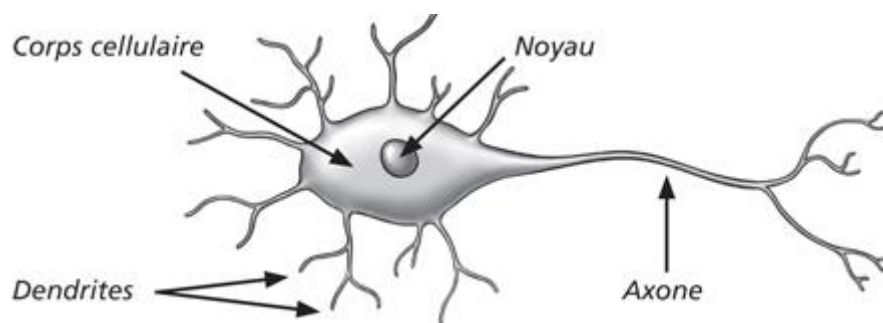


Figure 12 : Neurone biologique

Le système nerveux est composé de milliards de cellules : c'est un réseau de neurones biologiques. En effet, les neurones ne sont pas indépendants les uns des autres, ils établissent entre eux des liaisons et forment des réseaux plus ou moins complexes.

Le neurone biologique est composé de trois parties principales :

- **Le corps cellulaire** : composé du centre de contrôle traitant les informations reçues par les dendrites.
- **Les dendrites** : sont les principaux fils conducteurs par lesquels transitent l'information venue de l'extérieur.
- **L'axone** : est fil conducteur qui conduit le signal de sortie du corps cellulaire vers d'autres neurones.

Quant aux synapses, elles font effet de liaison et de pondération entre neurones et permettent donc aux neurones de communiquer entre eux.

Donc les neurones biologiques disposent d'un centre de contrôle (appelé cellule somatique) faisant la somme des informations recueillies par les dendrites. Ensuite, le centre de contrôle retourne un potentiel d'action en suivant les règles suivantes :

- Si la somme en entrée ne dépasse pas le seuil d'excitation : pas de message nerveux via l'axone.
- Si la somme en entrée dépasse le seuil d'excitation : un message nerveux est émis via l'axone.

3.2.1 Réseaux neurones artificiels :

Le réseau de neurones artificiel est basé sur le même modèle que le réseau de neurones biologique.

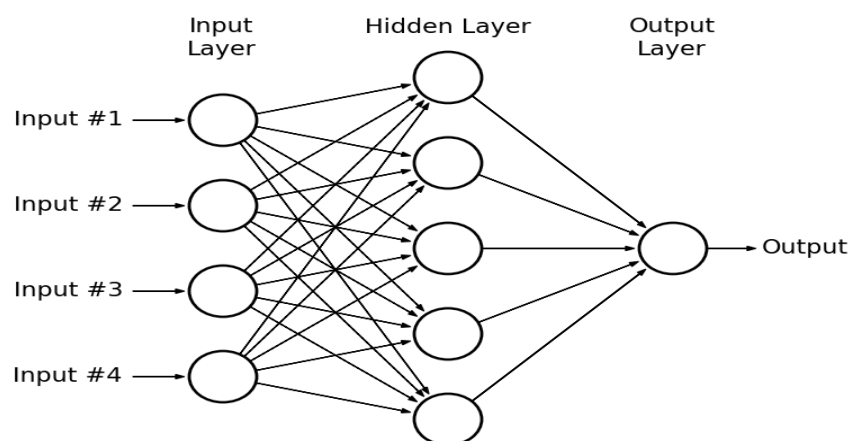


Figure 13 : Réseau de Neurone Artificiel

Ce réseau de neurones est composé des composants principaux suivants : [5]

- **Neurones : ensemble de fonction**

Ils prennent une donnée d'entrée et produisent une donnée de sortie. Un certain nombre de neurones sont groupés en couches (ou layers). Tous les neurones du même groupe remplissent un type de fonction similaire.

- **Couches : groupement de neurones**

Les **couches** (ou **layers**) contiennent des neurones et aident à faire circuler l'information. Il existe au moins deux couches dans un réseau de neurones : la couche d'entrée (input layer) et la couche de sortie (output layer).

- **Poids et biais : valeurs numériques**

Les **poids** et biais sont des variables du modèle qui sont mises à jour pour améliorer la précision du réseau. Un poids est appliqué à l'entrée de chacun des neurones pour calculer une donnée de sortie. Les réseaux de neurones mettent à jour ces poids de manière continue. Il existe donc une boucle de rétroaction mise en œuvre dans la plupart des réseaux de neurones.

Les biais sont également des valeurs numériques qui sont ajoutées une fois que les poids sont appliqués aux valeurs d'entrée. Les poids et les biais sont donc en quelque sorte des valeurs d'auto-apprentissage de nos réseaux de neurones.

3.2.2 Fonctionnement d'un réseau neuronal artificiel

Par analogie avec la biologie, un neurone artificiel reçoit l'information provenant des entrées x_i ($i=1, 2, 3, \dots, n$) par l'intermédiaire des connexions dont on affecte à chacune d'elles un poids w_i (abréviation de Wight en anglais) pondérant l'information.

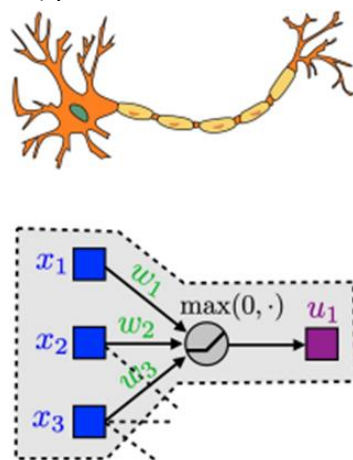


Figure 14 : Correspondance entre neurone artificiel et neurone biologique

Le neurone artificiel fonctionne en deux étapes :

- La première phase représente les prétraitements des données reçus en calculant le potentiel v_j des neurones
- Dans la deuxième phase, une fonction de transfert f appelée également **fonction d'activation**, calcule la valeur de l'état interne de neurone j à partir de la valeur potentiel v_j , cette valeur désigne la sortie de neurone :

$$z = f(b + x \cdot w) = f \left(b + \sum_{i=1}^n x_i w_i \right)$$

3.2.3 Fonction d'activation :

Comme son nom l'indique, la fonction d'activation est une formule mathématique (algorithme) activée dans certaines circonstances. Lorsque les neurones calculent la somme pondérée des valeurs d'entrée + le biais, elles sont transmises à la fonction d'activation, qui vérifie si la valeur calculée est supérieure au seuil requis. Si la valeur calculée est supérieure au seuil requis, la fonction d'activation est activée et une valeur de sortie est calculée. Cette valeur de sortie est ensuite transmise aux couches suivantes ou précédentes (en fonction de la complexité du réseau), ce qui peut aider les réseaux de neurones à modifier le poids de leurs neurones.

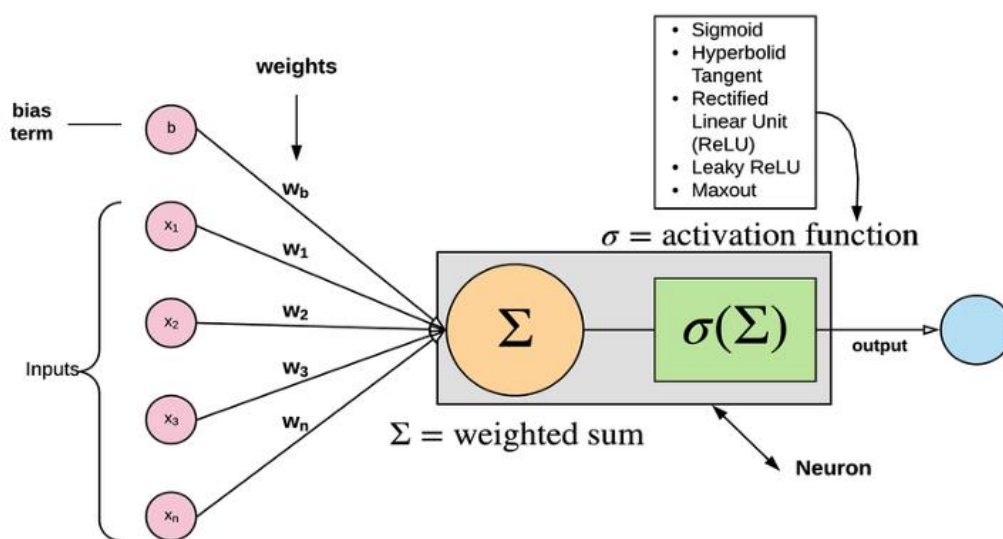


Figure 15 : Géométrie du neurone artificiel

Selon le problème à résoudre (classification, régression,) on utilise des fonctions différentes. Pour choisir la bonne fonction d'activation il faut à la fois considérer transformation directe qu'elle applique aux données mais aussi à sa dérivée qui sera utilisée pour ajuster les poids lors de la rétro propagation.

Il existe plusieurs variantes de fonctions d'activation parmi lesquelles :

- **Rectified Linear Unit (ReLU)**

La fonction **Rectified Linear Unit (ReLU)** est la fonction d'activation la plus simple et la plus utilisée. Elle donne x si x est supérieur à 0, 0 sinon. Autrement dit, c'est le maximum entre x et 0 :

$$f(x) = \max(0, x)$$

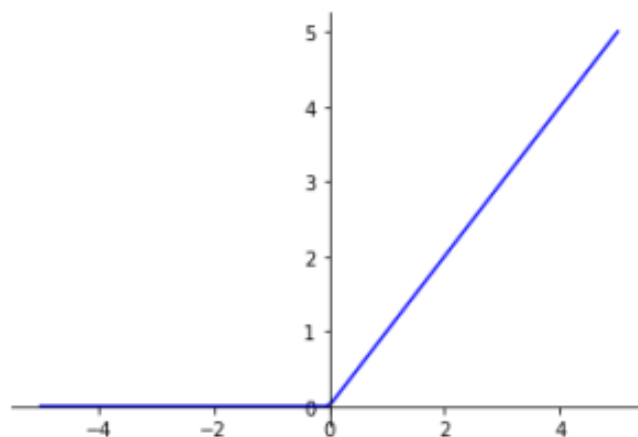


Figure 16 : Fonction ReLU-Rectified Linear

Cette fonction permet d'effectuer un filtre sur nos données. Elle laisse passer les valeurs positives ($x > 0$) dans les couches suivantes du réseau de neurones.

Elle est utilisée presque partout mais surtout pas dans la couche finale, elle est utilisée dans les couches intermédiaires.

- **Sigmoid**

La fonction Sigmoid donne une valeur entre 0 et 1, une probabilité. Elle est donc très utilisée pour la classification binaire, lorsqu'un modèle doit déterminer seulement deux labels.

$$f(x) = \frac{1}{1 + e^{-x}}$$

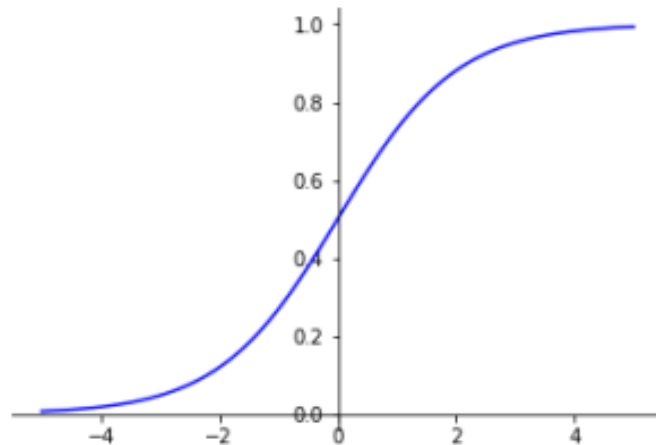


Figure 17 : Fonction Sigmoid

- **Softmax**

La fonction Softmax permet de transformer un vecteur réel en vecteur de probabilité. On l'utilise souvent dans la couche finale d'un modèle de classification, notamment pour les problèmes multi classe :

$$f(z_i) = \frac{e^{z_i}}{\sum_j e^{z_j}}$$

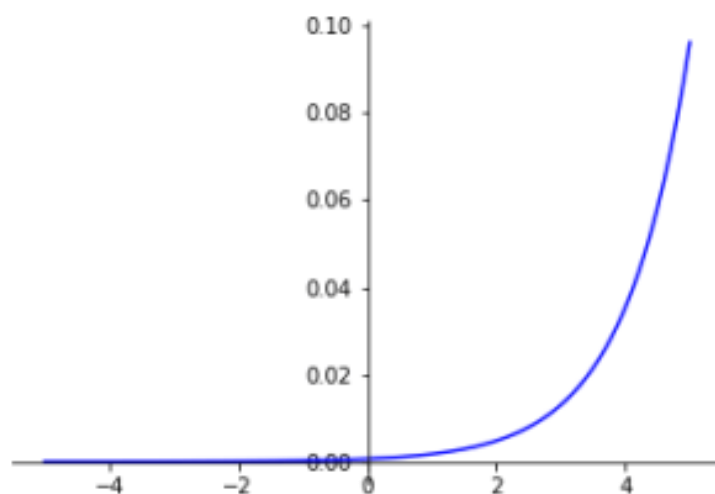


Figure 18 : Fonction Softmax

➤ Quelle fonction pour quel cas ?

Comme on l'a vu précédemment, plusieurs fonctions d'activation peuvent être utilisées dans un modèle selon le choix de l'utilisateur. Ainsi, il faudra choisir la bonne fonction d'activation en fonction du type de problème auquel on a fait :

Type de problème	Fonction d'activation de la dernière couche
Classification Binaire	Sigmoid
Classification multi classe, à label unique	Softmax
Classification multi classe, multi label	Sigmoid

Tableau 2 Fonction d'activation Vs Modèle

3.2.4 Apprentissage des réseaux neurones :

On appelle apprentissage des réseaux de neurones la procédure qui consiste à estimer les paramètres des neurones du réseau, afin que celui-ci remplisse au mieux sa tâche. L'apprentissage est la phase qui permet de déterminer les poids et le seuil de chaque neurone dans un réseau de neurones artificiel. Il a comme objectif l'amélioration des performances futures du réseau, sur la base de connaissance acquise au fur et à mesure. Une fois l'apprentissage terminé, le réseau est prêt à remplir sa fonction. Cette fonction peut être la classification, l'optimisation etc... [6]

3.2.4.1 L'algorithme de la retro-propagation du gradient d'erreur :

L'algorithme de la retro-propagation altère les coefficients synaptiques de réseau dans le sens inverse du gradient du critère d'erreur, en utilisant seulement les données d'entrée/sortie. A chaque itération, on retire un exemple d'apprentissage et on calcule une nouvelle estimation du poids synaptique. Cette itération consiste en deux phases : [7]

- **Propagation :** à chaque itération, un élément de l'ensemble d'apprentissage est introduit à travers la couche d'entrée. L'évaluation des sorties du réseau se fait couche par couche, de l'entrée vers la sortie.
- **Retro-propagation :** cette étape est similaire à la précédente. Cependant, les calculs s'effectuent dans le sens inverse (de la sortie vers l'entrée).

À la sortie du réseau, on forme le critère de performance E en fonction de la sortie réelle de système et sa valeur désirée. Puis, on évalue le gradient de E par rapport aux différents poids en commençant par la couche de sortie et en remontant vers la couche d'entrée.

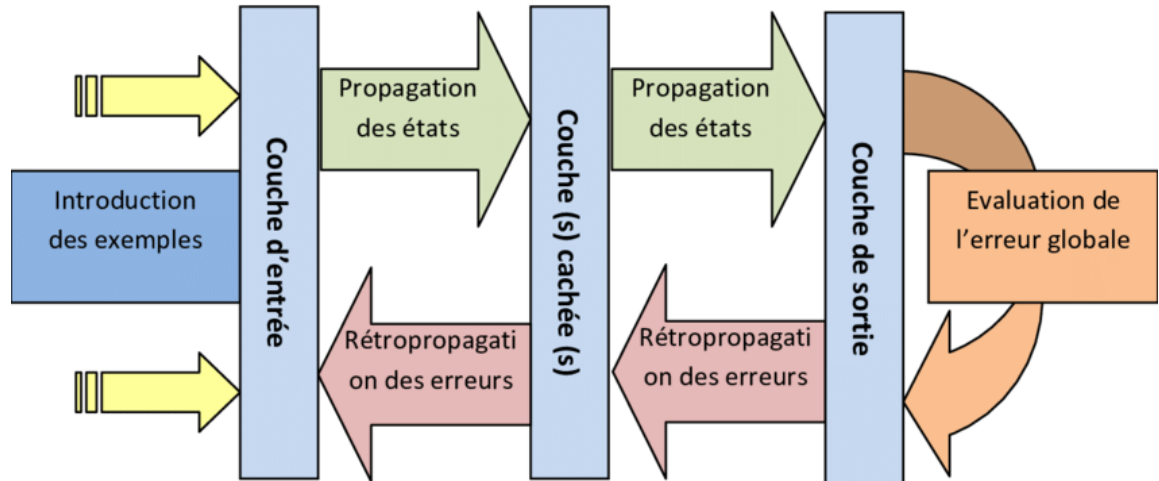


Figure 19 : Apprentissage par rétro propagation

3.2.4.2 Fonction de perte (Loss function) :

Une fonction de perte, ou Loss function, est une fonction qui évalue l'écart entre les prédictions réalisées par le réseau de neurones et les valeurs réelles des observations utilisées pendant l'apprentissage. Plus le résultat de cette fonction est minimisé, plus le réseau de neurones est performant. Sa minimisation, c'est-à-dire réduire au minimum l'écart entre la valeur prédite et la valeur réelle pour une observation donnée, se fait en ajustant les différents poids du réseau de neurones. [8]

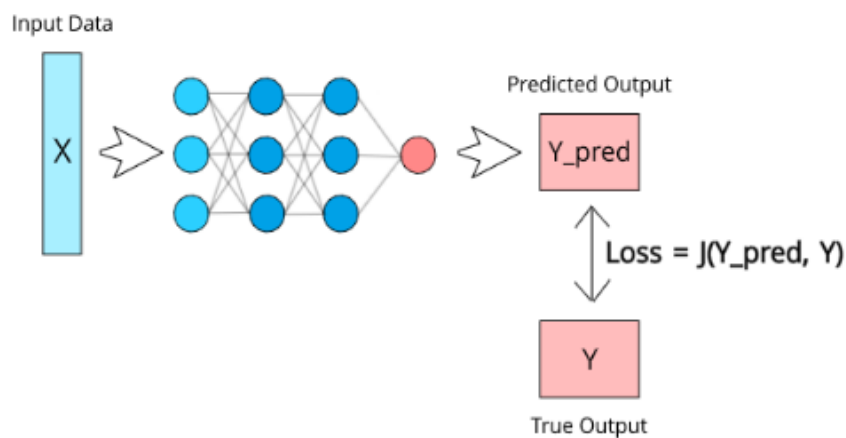


Figure 20 : Modélisation de la fonction de perte

Il n'y a pas une seule fonction de perte qui fonctionne pour tous les types de données. Cela dépend d'un certain nombre de facteurs, notamment la présence de valeurs aberrantes, le choix de l'algorithme d'apprentissage automatique, l'efficacité temporelle de la descente de gradient et la facilité de recherche des dérivées. Quant à nous on a utilisé la fonction :

- **Categorical cross-entropy :**

C'est une fonction de perte qui est utilisée pour la catégorisation à étiquette unique. C'est le cas lorsqu'une seule catégorie est applicable pour chaque point de données. En d'autres termes, un exemple ne peut appartenir qu'à une seule classe.

$$L(y, \hat{y}) = - \sum_{j=0}^M \sum_{i=0}^N (y_{ij} * \log(\hat{y}_{ij}))$$

3.2.4.3 Optimisation par le gradient descend :

Le **gradient descend** est un algorithme d'optimisation qui permet de calculer le minimum local (point orange) d'une fonction convexe (fonction de perte) en changeant au fur et à mesure (itérations) les paramètres de cette fonction.

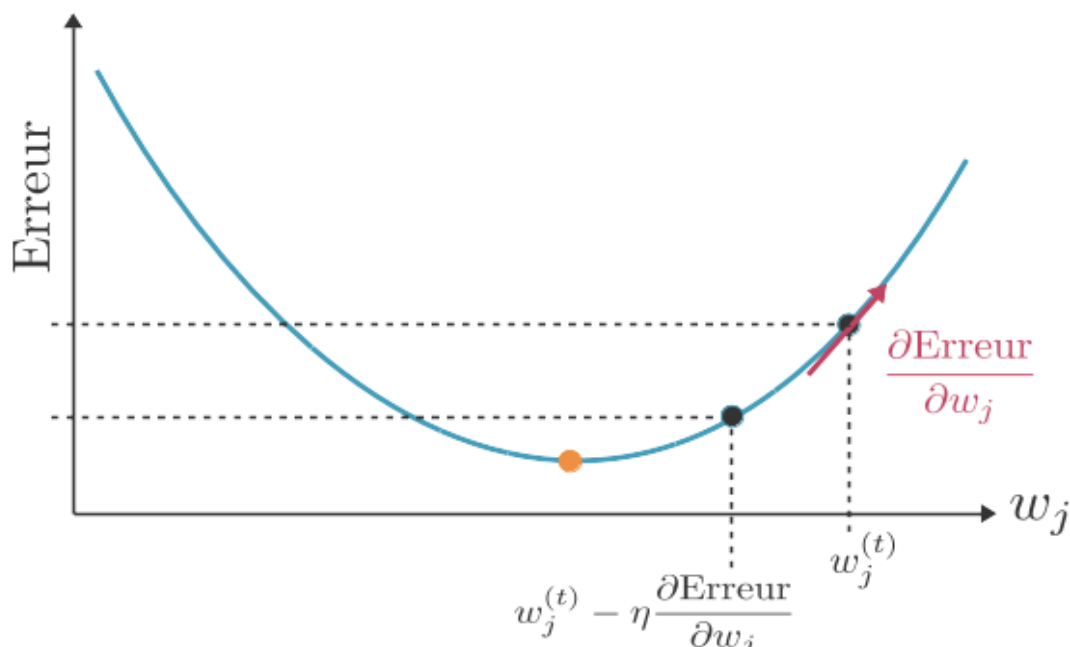


Figure 21 : Minimum local de la fonction de perte

En d'autres termes, le gradient descend est un algorithme permettant de trouver le minimum local d'une fonction différentiable. La descente de gradient est simplement utilisée pour trouver des valeurs aux paramètres (poids w_j) d'une fonction permettant d'atteindre ce minimum local. Ainsi, les valeurs initiales des poids de connexion sont choisies aléatoirement, ensuite après chaque observation il y aura une application à chacune des poids la règle de mise à jour suivante :

$$w_j^{(t+1)} = w_j^{(t)} - \eta \frac{\delta_{Erreur}}{\delta w_j}$$

Il existe trois variantes de l'algorithme de la descend de gradient :

- **Batch gradient descend (BGD)**

Calcule l'erreur pour chaque exemple de l'ensemble de données d'apprentissage, mais ne met à jour le modèle qu'après l'évaluation de tous les exemples d'apprentissage.

- **Stochastique gradient descend (SGD)**

Calcule l'erreur et met à jour le modèle pour chaque exemple de l'ensemble de données d'apprentissage.

- **Mini-Batch gradient descend :**

Divise le jeu de données d'apprentissage en petits lots qui sont utilisés pour calculer l'erreur du modèle et mettre à jour les coefficients du modèle.

3.2.5 Les Types de réseau de neurones :

Il existe deux types de réseau neurones :

- **Les réseaux de neurones feed-forwarded :**

Feed-forwarded (propagation avant) signifie que la donnée traverse le réseau d'entrée à la sortie sans retour en arrière de l'information. [9]

Typiquement, dans la famille des réseaux à propagation avant, on distingue les réseaux monocouches (perception simple) et les réseaux multicouches (perception multicouche). La perception simple est dite simple parce qu'il ne dispose que de deux couches : la couche en entrée et la couche en sortie. Le réseau est déclenché par la réception d'une information en entrée. Le traitement de données dans ce réseau se fait entre la couche d'entrée et la couche de sortie qui sont toutes reliées entre elles. Le réseau intégral ne dispose ainsi que d'une matrice de poids.

Le fait de disposer d'une seule matrice de poids limite le perceptron simple à un classificateur linéaire permettant de diviser l'ensemble d'informations obtenues en deux catégories distingue.

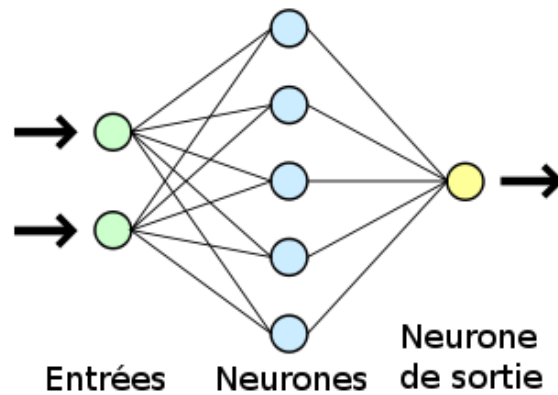


Figure 22 : Perception simple

Le perceptron multicouche se structure de la même façon. L'information entre par une couche d'entrée et sort par une couche de sortie. A la différence du perceptron simple, le perceptron multicouche dispose entre la couche en entrée et la couche en sortie une ou plusieurs couche dites « cachées ». Le nombre de couches correspond aux nombres de matrices de poids dont dispose le réseau. Un perceptron multicouche est donc mieux adapté pour traiter les types de fonctions non-linéaires. [10]

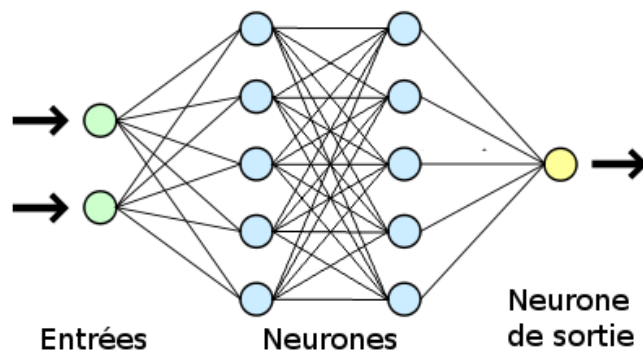


Figure 23 : Perception Multicouche

Pour le traitement d'informations complexes et très variées, il est envisageable de créer plusieurs réseaux de neurones distincts dédiés à traiter chacun une partie de l'information.

Ces réseaux de neurones sont appelés des réseaux neuronaux convolutifs (Convolutional Neural Networks). Ces réseaux peuvent être imaginés comme une compilation d'un segment d'informations pour au final traiter l'ensemble de l'information (par exemple le traitement d'image, de vidéos, de textes).

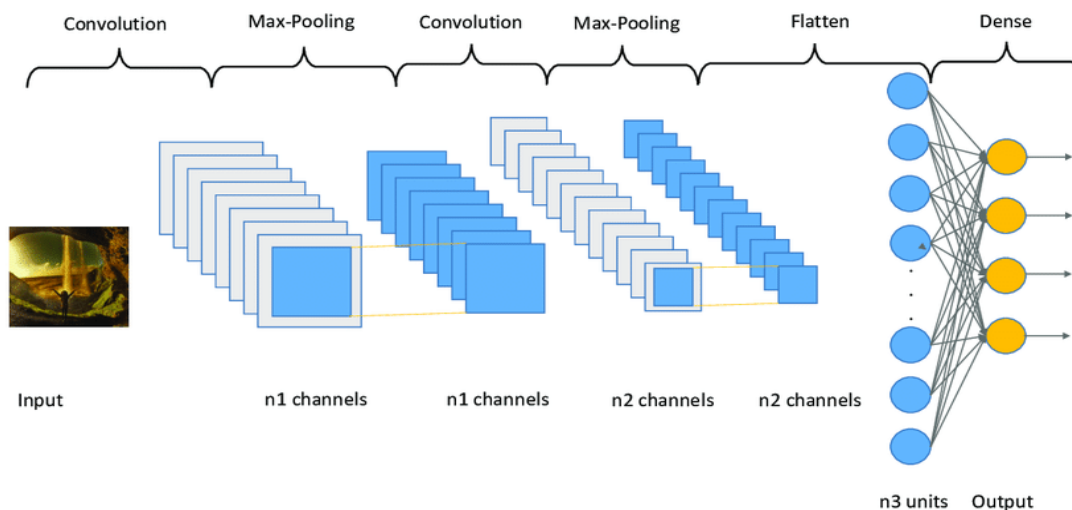


Figure 24 : Réseau neuronal convolutif

- **Les réseaux de neurones récurrents :**

Les réseaux de neurones récurrents traitent l'information en cycle. Ces cycles permettent au réseau de traiter l'information plusieurs fois en la renvoyant à chaque fois au sein du réseau. La force réseaux de neurones récurrents réside dans leur capacité de prendre en compte les informations contextuelles suite à la récurrence du traitement de la même information. Cette dynamique auto-entretient le réseau.

Les réseaux de neurones récurrents à couche multiples revendiquent quant à eux la particularité de posséder des couples (entrée / sortie) comme les perceptrons entre lesquels la donnée véhicule à la fois en propagation en avant et en rétro propagation.

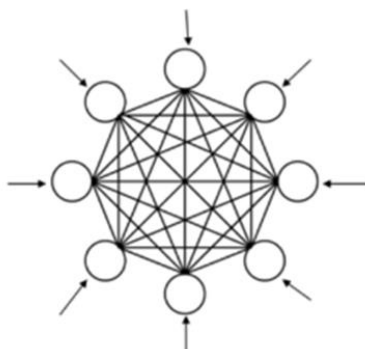


Figure 25 : Réseau neuronal récurrent

3.3 Traitement Numérique du langage :

3.3.1 Définition :

Le NLP ou le Traitement numérique du Langage est une discipline qui porte essentiellement sur la compréhension, la manipulation et la génération du langage naturel par les machines. Ainsi, Le NLP est réellement à l'interface entre la science informatique et linguistique. Il porte donc sur la capacité de la machine à interagir directement avec l'humain.

3.3.2 NLP et Chatbots :

Les méthodes NLP sont au cœur du fonctionnement des Chatbots actuels. Bien que ces systèmes ne soient pas totalement parfaits, ils peuvent aujourd'hui facilement gérer des tâches standards telles renseigner des clients sur des produits ou services, répondre à leurs questions, etc. Ils sont utilisés par plusieurs canaux, dont l'Internet, les applications et les plateformes de messagerie.

3.3.3 Principaux challenges :

Les règles qui régissent la transformation de texte en langage naturel en information ne sont pas faciles à comprendre pour les ordinateurs. Parmi les principaux challenges :

- **Ambiguïté :**

En langage naturel, les mots sont uniques mais peuvent avoir des significations différentes selon le contexte, ce qui entraîne une ambiguïté au niveau lexical, syntaxique et sémantique.

- **Synonymie :**

Un autre phénomène clé en langage naturel est le fait que nous pouvons exprimer la même idée avec différents termes qui dépendent également du contexte spécifique.

- **Coréférence :**

Les tâches de coréférence consistent à trouver toutes les expressions qui se réfèrent à la même entité. Il s'agit d'une étape importante pour de nombreuses tâches de NLP de haut niveau qui impliquent une compréhension du texte dans sa globalité, telles que le résumé de documents, la réponse à des questions et l'extraction d'informations. Ce problème a connu un renouveau avec l'introduction de techniques de pointe de Deep Learning.

- **Bag of Words (sac de mots):**

Le sac de mots est l'un des algorithmes les plus fondamentaux du traitement du langage naturel (NLP). Son travail principal est de trouver des informations à partir de données. Il est principalement utilisé pour la classification de documents, à savoir la modélisation linguistique et la modélisation thématique.

Dans la plupart des cas nous rencontrons des données textuelles très désordonnées et non organisées. Les algorithmes d'apprentissage automatique ne peuvent pas travailler directement sur les données brutes et les données textuelles, car il s'agit d'une machine qui demande la saisie de nombres en entrée au lieu de texte. Nous devons donc convertir tout le texte en nombres ou en vecteur pour modéliser notre algorithme.

3.3.4 Etape de BOW :

Pour extraire les caractéristiques d'un document, il faut suivre les étapes mentionnées ci-dessus :



Figure 26 : Etapes de BOW

- **Prétraitement du texte :**

- **Nettoyage :** Variable selon la source des données, cette phase consiste à réaliser des tâches telles que la suppression d'url, d'emoji, etc.
- **Stemming :** un même mot peut se retrouver sous différentes formes en fonction du genre (masculin féminin), du nombre (singulier, pluriel), la personne (moi, toi, eux...) etc. Le stemming désigne généralement le processus heuristique brut qui consiste à découper la fin des mots dans afin de ne conserver que la racine du mot.



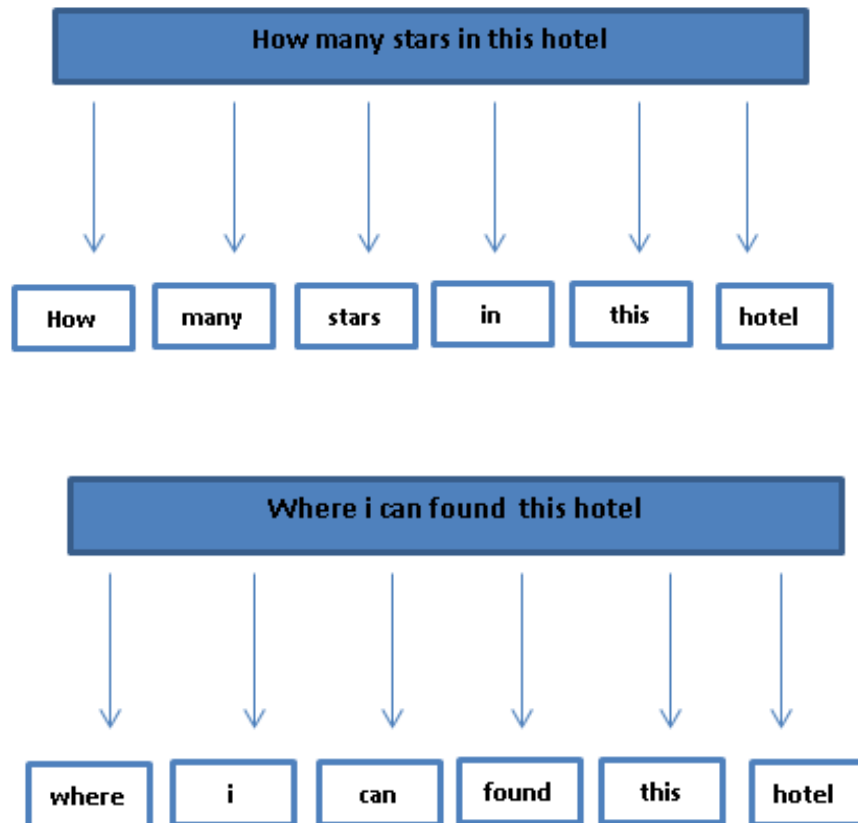
- **Lemmatisation** La lemmatisation permet de supprimer uniquement les terminaisons inflexibles et donc à isoler la forme canonique du mot, connue sous le nom de lemme.



- **Autres opérations :** suppression des chiffres, ponctuation, symboles et stopwords, passage en minuscule.

- **Présentation numérique des données :**

- **Tokenisation :** La tokenisation est une étape simple mais indispensable. Le principe est simple il faut découper la ou les phrases en mots.



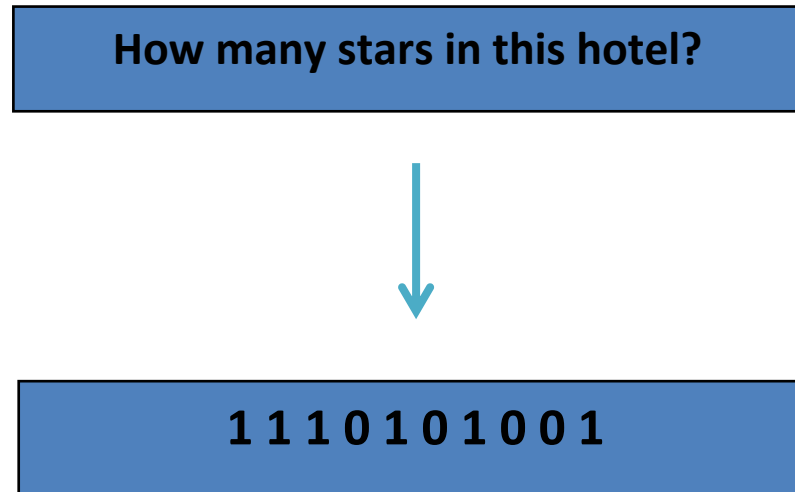
- **Création de vocabulaire :**

Cette étape sert à construire un dictionnaire qui regroupe tous les mots unique (token) qui constituent Les phrases de notre jeu de données.



- **Encodage :**

Afin de pouvoir appliquer les méthodes de Machine Learning aux problèmes relatifs au langage naturel, il est indispensable de transformer les données textuelles en données numériques.



3.3.5 Limite de BOW :

L'efficacité de ces méthodes diffère selon le cas d'application. Toutefois, elles présentent deux principales limites :

- Plus le vocabulaire du corpus est riche, plus la taille des vecteurs est grande, ce qui peut représenter un problème pour les modèles d'apprentissage.
- Le comptage d'occurrence des mots ne permet pas de rendre compte de leur agencement et donc du sens des phrases.

3.4 Conclusion :

Dans cette partie on a présenté le concept de réseau de neurones et comment il fonctionne, et on a introduit les types de ces réseaux les plus populaires. De plus nous avons représenté un aperçu sur la manière d'apprentissage de ces réseaux et les composants qui les rendent artificiellement intelligents.

Chapitre 4

Préparation de l'environnement de travail

Dans ce chapitre on va présenter les différents outils de programmation et du développement qu'on a utilisé pour réaliser notre projet.

4.1 Python :



Python est un langage de programmation libre (open source) créé par le programmeur *Guido Van Rossum* en 1991. Il tire son nom de l'émission *Monty Python's Flying Circus*. [11]

Il s'agit d'un langage de programmation interprété qui ne nécessite donc pas d'être compilé pour fonctionner contrairement à des langages comme le C et le C++.

Il fonctionne sous un grand nombre de plateformes incluant Linux, Windows, Mac OS X, OS/2, Android ainsi que les nombreuses variantes dérivées d'Unix. Il est orienté objet, c'est-à-dire qu'il est possible de concevoir en python des entités qui miment celles du monde réel avec un certain nombre de règles de fonctionnement et d'interaction.

Le plus intéressant sur ce langage est qu'il est gratuit et relativement simple à prendre en main, il permet aux programmeurs de se focaliser sur ce qu'ils font plutôt que sur la façon dont ils le font, ainsi dans un temps plus réduit que d'autre langage.

Ce langage de programmation de haut niveau est souvent utilisé par les développeurs dans :

- La création de service Web
- La programmation d'applications
- La génération de code
- La métaprogrammation

Il existe deux versions de Python : **Python 2 et Python 3**. Les différences entre ces deux versions sont multiples. Python 2.x est l'ancienne version qui continuera de recevoir des mises à jour officielles jusqu'en 2020 après elle continuera de subsister de façon non officielle. Python 3.x est la version actuelle du langage, elle apporte de nombreuses fonctionnalités nouvelles et très utiles, telles qu'un meilleur contrôle de concurrence et un interpréteur plus efficace.

Dans notre projet on a utilisé la version Python 3.

4.2 Flask :

4.2.1 Framework :

Un Framework ou infrastructure logicielle en français désigne en programmation informatique un ensemble d'outils et de composants logiciels à la base d'un logiciel ou d'une application. C'est le Framework encore appelé structure logicielle, canevas ou socle d'application en français qui établit des fondations d'un logiciel ou son squelette applicatif.

L'objectif du Framework est de simplifier et d'uniformiser le travail des développeurs. Il fonctionne comme un cadre ou un patron mais son maniement suppose d'avoir déjà un profil expérimenté. En général, une infrastructure logicielle est associée spécifiquement à un langage de script ou de programmation. Par exemple Flask est un Framework pour Python.

L'utilisation d'un Framework présente de nombreux avantages car il instaure en quelque sorte sa « ligne de conduite ». Tous les développeurs Flask codent de façon assez homogène (leurs codes ont la même structure, les mêmes principes). De ce fait, lorsqu'un développeur rejoint un projet utilisant un Framework qu'il connaît déjà, il comprendra très vite son architecture et pourra se mettre rapidement au travail. [12]

Il existe différents types de Framework :

- **Framework d'infrastructure système** : Pour développer des systèmes d'exploitation, des interfaces graphiques, des outils de communication.
- **Framework d'intégration intergicielle (middleware)** : Pour fédérer des applications hétérogènes et mettre à disposition différentes technologies sous la forme d'une interface unique.
- **Framework d'entreprises** : Pour développer des applications spécifiques au secteur d'activité de l'entreprise.
- **Framework de gestion de contenu** : Pour la création, la collecte, le classement, le stockage et la publication de biens numérisés.

L'utilité d'un Framework lors du développement d'un projet Web se situe à plusieurs niveaux :

- **La rapidité** : une base de travail existe déjà, donc le développeur web n'a pas besoin de partir de zéro pour créer votre site web.

- **Flexibilité** : vous pouvez choisir d'utiliser ou non certains composants du Framework pour améliorer le référencement naturel de votre site.
- **Architecture** : en utilisant un bon Framework, vous avez du code propre et fonctionnel qui ne ralentit pas le fonctionnement du site.
- **Productivité** : que ce soit un développement en solo ou en équipe, un Framework est un outil puissant puisque tout est parfaitement organisé.
- **Communauté** : vous bénéficiez de l'appui de toute une communauté en ligne (support et forum) qui vous aidera à corriger les bugs ou résoudre des problèmes de programmation.

4.2.2 Flask :



Flask est un micro Framework open-source de développement web écrit en Python. Il est classé comme micro Framework car il est conçu pour que le cœur de l'application reste simple et évolutif. Au lieu d'une couche d'abstraction pour la prise en charge de la base de données, Flask prend en charge les extensions pour ajouter de telles fonctionnalités à l'application. [13]

Flask a été créé en 1er avril 2010 par Armin Ronacher de Poccoo, un groupe international de passionnés de Python formé en 2004. Selon Ronacher, l'idée était à l'origine un poisson d'avril assez populaire pour en faire une application sérieuse.

Contrairement au Framework Django, Flask est très Pythonique. Il est facile de démarrer avec Flask, car il n'a pas une courbe d'apprentissage énorme. De plus, il est très explicite, ce qui augmente la lisibilité.

Flask se base sur deux modules Werkzeug et Jinja2 pour proposer plusieurs des fonctionnalités suivantes :

- Serveur de développement et debugger.
- Simplifie l'écriture de tests unitaires.
- Moteur de Template pour le rendu HTML.
- Supporte les cookies sécurisés (session).
- Entièrement compatible avec WSGI 1.0.

4.3 HTML & CSS :

4.3.1 HTML :

HTML



Le **HTML** ou l'**HyperText Markup Language** désigne un type de langage informatique descriptif. Il s'agit plus précisément d'un format de données utilisé dans l'univers d'Internet pour la mise en forme des pages Web. Il permet, entre autres, d'écrire de l'hypertexte, mais aussi d'introduire des ressources multimédias dans un contenu. Développé par le **W3C** (World Wide Web Consortium) et le **WHATWG** (Web Hypertext Application Technology Working Group), le format ou langage **HTML** est apparu dans les années 1990. Il a progressivement subi des modifications et propose depuis 2014 une version **HTML5** plus aboutie. [14]

L'HTML permet à un créateur de sites Web de gérer la manière dont le contenu de ses pages Web va s'afficher sur un écran, via le navigateur. Il repose sur un système de balises permettant de titrer, sous-titrer, mettre en gras, etc., du texte et d'introduire des éléments interactifs comme des images, des liens, des vidéos... L'HTML est plus facilement compris des robots de crawl des moteurs de recherche que le langage JavaScript, aussi utilisé pour rendre les pages plus interactives.

4.3.2 CSS :

CSS



Le **CSS** ou le **Cascading Style Sheets**, est un langage informatique utilisé sur Internet pour la mise en forme de fichiers et de pages HTML. On le traduit en français par feuilles de style en cascade. Apparu dans les années 1990, le **CSS** se présente comme une alternative à la mise en forme via des balises, notamment HTML. Un peu plus complexe à maîtriser, il permet un gain de temps considérable dans la mise en forme d'une page web par rapport à ces balises. Grâce au **CSS**, vous pouvez en effet appliquer des règles de mise en forme (titrage, alignement, polices, couleurs, bordures, etc.) à plusieurs documents simultanément. [15]

Le but de CSS est de séparer la structure d'un document HTML et sa présentation. En effet, avec HTML, on peut définir à la fois la structure (le contenu et la hiérarchie entre les différentes parties d'un document) et la présentation. Mais cela pose quelques problèmes.

Avec le couple HTML/CSS, on peut créer des pages web où la structure du document se trouve dans le fichier HTML tandis que la présentation se situe dans un fichier CSS.

4.4 JavaScript :



JavaScript désigne un langage de développement informatique, et plus précisément un **langage de script orienté objet**. On le retrouve principalement dans les pages Internet. Il permet, entre autres, d'introduire sur une page web ou HTML des petites animations ou des effets. Créé en 1995 par Brendan Eich, en même temps que la technologie Java, le langage JavaScript se distingue des langages serveurs par le fait que l'exécution des tâches est opérée par le navigateur lui-même, sur l'ordinateur de l'utilisateur, et non sur le serveur web. Il s'active donc généralement sur le poste client plutôt que côté serveur. [16]

4.5 Conclusion :

Tous ces outils et ces langages nous ont servis à réaliser un chatbot performant avec une interface simple et utile.

Chapitre 5

Réalisation

5.1 Base de données :

5.1.1 JSON :



JavaScript Object Notation (JSON) est un format texte standard pour représenter des données structurées basées sur la syntaxe d'objet JavaScript. Il est couramment utilisé pour transmettre des données dans des applications Web (par exemple, envoyer des données du serveur au client, afin qu'elles puissent être affichées sur une page Web, ou vice versa). [17]

Le choix d'utiliser JSON pour notre base de données vient du fait que son format est compréhensible par tous (humain et machine), aucun apprentissage n'est requis puisque la syntaxe n'utilise que quelques marques de ponctuations, et ne dépend d'aucun langage (format d'échange de données ouvert). Comme ce format est très ouvert, il est pris en charge par de nombreux langages : JavaScript, PHP, Perl, Python, Ruby, Java, ... Aussi permet de stocker des données de différents types : chaînes de caractères (y compris des images en base64), nombres, tableaux (array), objets, booléens (true, false), la valeur null. En fin sa structure en arborescence et sa syntaxe simple lui permet de rester très léger et efficace.

5.1.2 La structure de la base de données :

```
"intents": [
  {
    "tag": "greeting",
    "patterns": ["Hi there", "How are you", "Is anyone there?", "Hey", "Hola", "Hello", "Good day", "Hey", "Hi"],
    "responses": ["Hi stranger", "hi yes, how can I help", "Ey watsup", "Hola", "hello"],
    "name": ""
  },
  ...
]
```

Notre base de données contient les éléments suivants :

- **Intents** : contient les catégories des choses qui peuvent être répondu par le chatbot
- **Tag** : les différents catégorises fournis avec du texte pour que le chatbot reconnaisse de quoi il est interrogé
- **Patterns** : ce sont les motifs ou questions que l'utilisateur pourrait poser au chatbot et ensuite ce dernier va prendre cela comme données d'entraînements et voir quelle catégorie appartient la question.

- **Responses** : les réponses statiques aux questions ou demandes de l'utilisateur.
- **Name** : l'identité de la catégorie, par exemple chaque hôtel a son propre identifiant pour le distinguer.

5.1.3 L'augmentation des données :

Data augmentation ou l'augmentation des données est la pratique consistant à synthétiser de nouvelles données à partir des données disponibles. Cela pourrait être appliqué à toute forme de données, des nombres aux images. Elle est utile pour améliorer les performances et les résultats des modèles d'apprentissage automatique en formant des exemples nouveaux et différents pour former des ensembles de données. Si l'ensemble de données dans un modèle d'apprentissage automatique est riche et suffisant, le modèle est plus performant et plus précis.

Il existe des techniques classiques et avancées d'augmentation de données pour la reconnaissance d'images et le traitement du langage naturel. [18]

Pour le traitement du langage naturel (NLP) :

Les techniques d'augmentation des données sont appliquées aux niveaux des caractères, des mots et du texte. Les méthodes courantes pour l'augmentation des données en NLP sont :

- Opération Easy Data Augmentation (EDA) : remplacement de synonymes, insertion de mots, échange de mots et suppression de mots.
- Traduction arrière.
- Incrustations de mots contextualisées. [19]

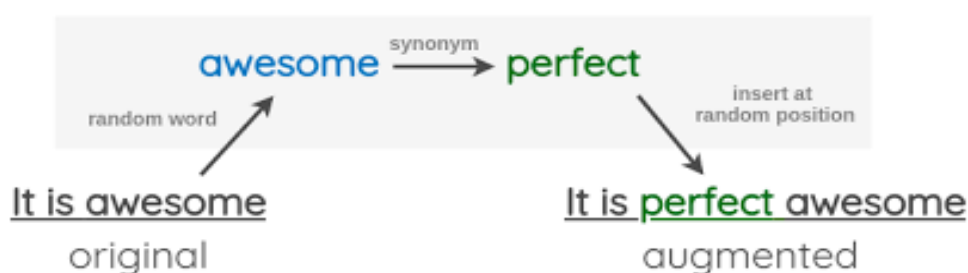


Figure 27 Fonctionnement d'un EDA

Dans ce projet on a utilisé un EDA pour l'enrichissement de notre base de données.

5.1.4 Chatito :



Chatito est un puissant langage spécifique à un domaine conçu pour simplifier le processus de création, d'extension et de maintenance d'ensembles de données pour la formation et la validation de modèles de traitement du langage naturel (NLP) pour la classification de texte, la reconnaissance d'entités nommées, le remplissage d'emplacements ou des tâches équivalentes. Il est caractérisé par : [20]

- **Rapidité** : générer des échantillons en les triant d'un nuage de probabilités à la demande
- **Simplicité** : devrait être compréhensible par un utilisateur qui travaille avec chatito pour la première fois.

```

findRestaurantsByCity.chatito x common.chatito x hotels.chatito x New file Generate Dataset
1 import ./common.chatito
2
3 # Ways to request a hotel within a location (using probability operator)
4
5
6 %[findHotelsByCity]('training': '100', 'testing': '100' 'distribution': 'even')
7 *[1]~[hi?] ~[please?] ~[find?] ~[hotels] ~[located at] ~[marrakech] ~[thanks?]
8 *[1]~[hotels] ~[located at] ~[marrakech]
9 *[1]~[What are] ~[the best] ~[hotel] ~[in marrakech]
10
11 ~[marrakech]
12 marrakech
13 this city
14 here
15 ~[located at]
16 in the area of the
17 near the
18 ~[hotels]
19 hotel
20 resort
21 motel
22 ~[like]
23 like
24 choise
  
```

Figure 28 Interface de Chatito

Dans cet exemple on a pris 3 façons de demander un hôtel dans la ville de Marrakech. Dans chaque phrase on trouve les entités alias qui sont défini par les symboles ~ [en début de ligne, suivis du nom de l'alias]. Les alias sont les mots qui vont être remplacé par leur synonyme pour enrichir la base de données (par exemple Marrakech va être remplacé par 'This city' et 'here').

L'expression *[1] est utilisé pour contrôler la probabilité de génération des phrases afin d'éviter que le générateur choisisse très probablement des phrases qui ont plus de combinaison possible et omettra certaines phrases qui peuvent être plus importantes dans l'ensemble de données. Dans cet exemple les trois phrases on la même probabilité d'être générées.

5.2 L'application Web :

A propos l'interface de notre application, nous avons essayé de la simplifier le plus possible afin de permettre aux utilisateurs de comprendre facilement comment elle fonctionne.

Comme il est présenté dans la figure 29 ci-dessous, on constate que l'interface présente en 1ère lieu les thèmes proposés par IRM Chatbot (Hotels, Restaurants, Places)

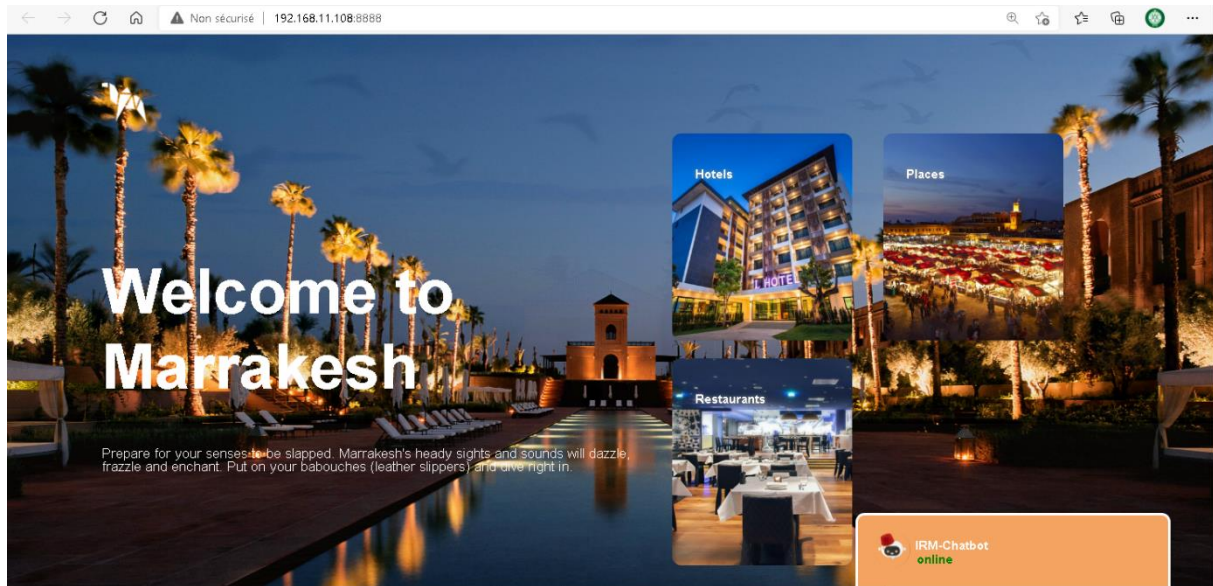
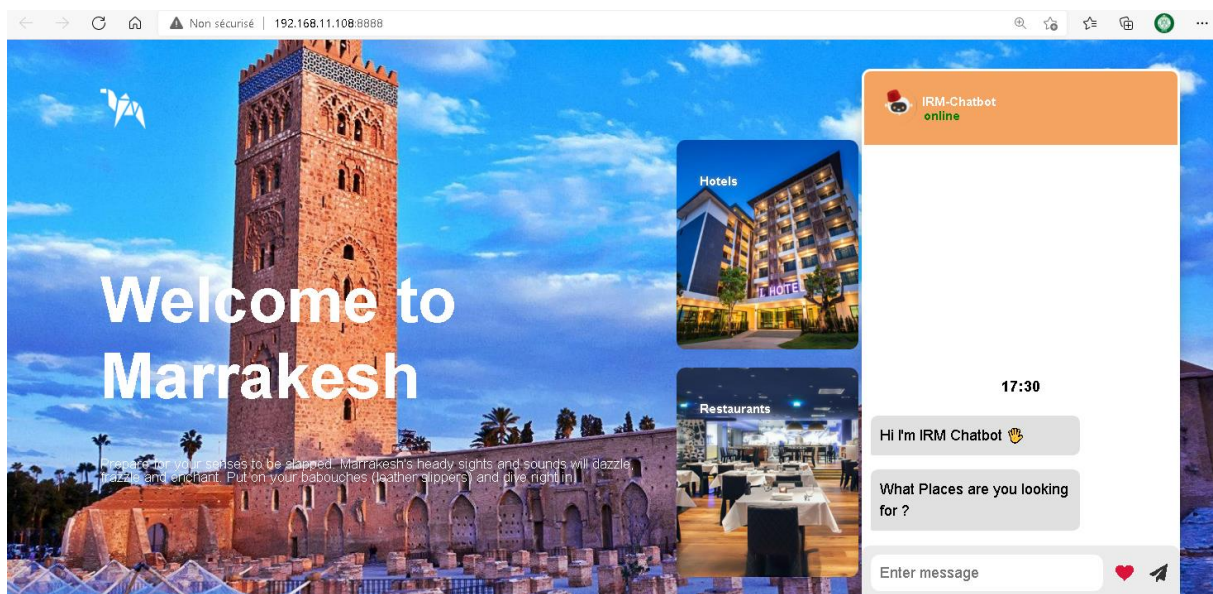


Figure 29 Interface d'accueil

En cliquant sur l'un des boutons (par exemple Place), la conversation de chat s'affiche comme suite :



Dans cette conversation, l'utilisateur peut poser n'importe quelle question dans le cadre des thèmes suggérés, la réponse du bot peut être sous forme du texte, bouton, image ou localisation par google maps.

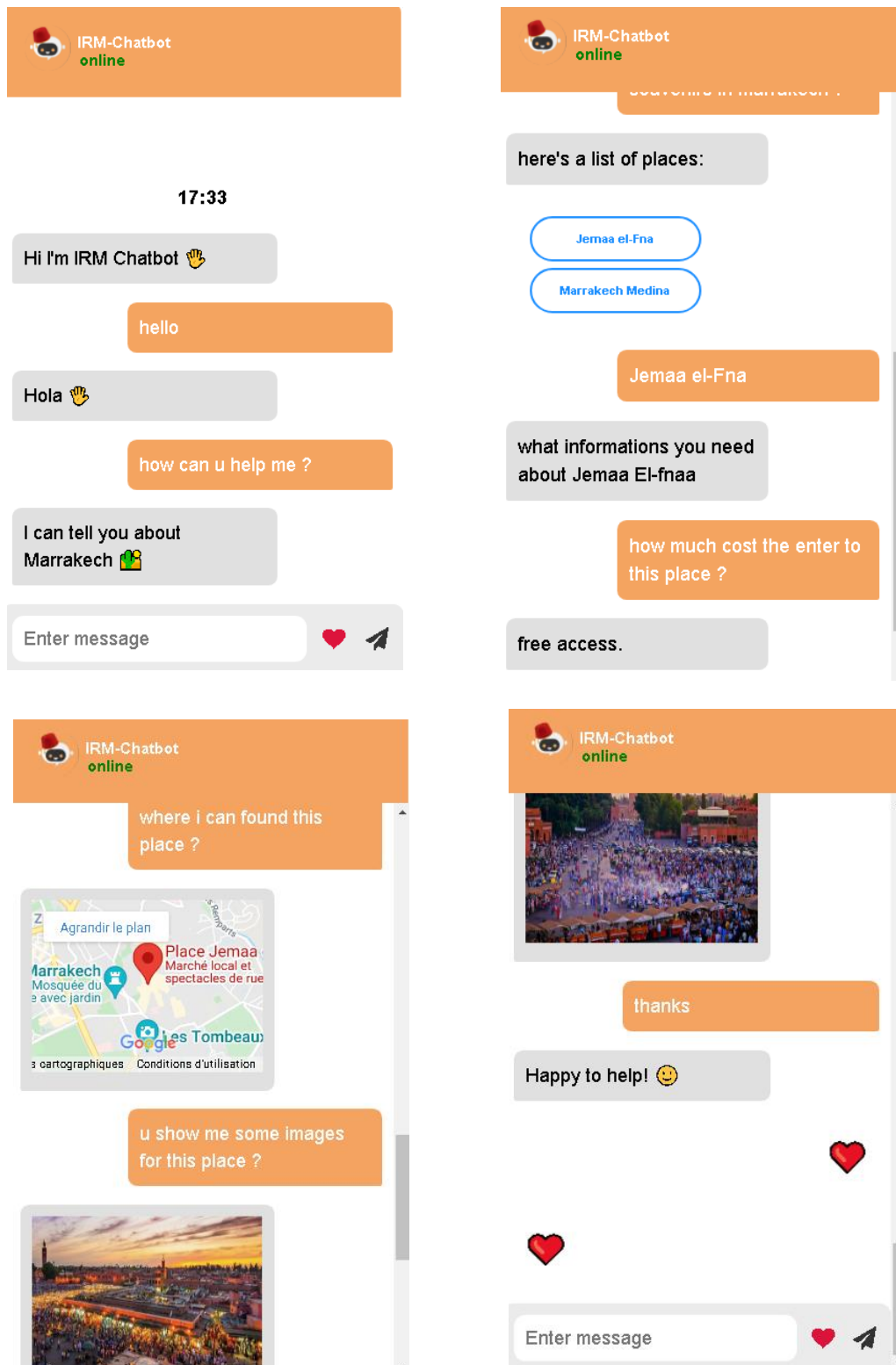


Figure 30 Interfaces de chat

Conclusion Générale :

Pour la génération de nos grands-parents, échanger de manière naturelle avec un robot était de la pure science-fiction, Ce n'était possible qu'au cinéma ou dans les récits imaginaires.

Pourtant, aujourd'hui pour la plupart des domaines, le chatbot est déjà le présent ou la future très proche de la relation client. Surtout le secteur du tourisme qui nécessite plus de communication avec les clients, c'est pourquoi ces interfaces de communication avec les utilisateurs sont si populaires et dépassent les applications mobiles dans ce secteur en optimisant davantage les tâches fournies par ces dernières et en offrant d'autres nouvelles possibilités et plus de flexibilité.

Durant la réalisation de ce projet on a pu approfondir les connaissances requises durant nos études universitaires surtout le secteur de Deep Learning et le traitement naturel de langage.

Notre application représente un assistant conversationnel pour la ville de Marrakech qui fournit aux visiteurs de cette ville des informations sur les hôtels, restaurants et les places touristiques avec certaines possibilités comme la réservation d'hôtel.

Références

- [1] [En ligne]. Available: <https://www.imagescreations.fr/qu-est-ce-qu-un-chatbot/>.
- [2] [En ligne]. Available: <https://www.monatourisme.fr/wp-content/uploads/2018/03/Note.pdf>.
- [3] [En ligne]. Available: <https://dzone.com/articles/understanding-architecture-models-of-chatbot-and-r>.
- [4] [En ligne]. Available: <http://www.chups.jussieu.fr/polys/histo/histoP1/POLY.Chp.7.2.html>.
- [5] [En ligne]. Available: <https://www.investopedia.com/terms/a/artificial-neural-networks-ann.asp>.
- [6] [En ligne]. Available: <https://www.universalis.fr/encyclopedie/reseaux-de-neurones-formels/5-l-apprentissage-des-reseaux-de-neurones-formels/>.
- [7] [En ligne]. Available: <https://dataanalyticspost.com/Lexique/retropropagation/>.
- [8] [En ligne]. Available: <https://machinelearningmastery.com/loss-and-loss-functions-for-training-deep-learning-neural-networks/>.
- [9] [En ligne]. Available: https://fr.wikipedia.org/wiki/R%C3%A9seau_de_neurones_%C3%A0_propagation_avant.
- [10] [En ligne]. Available: <https://www.analyticsvidhya.com/blog/2020/02/cnn-vs-rnn-vs-mlp-analyzing-3-types-of-neural-networks-in-deep-learning/>.
- [11] [En ligne]. Available: <https://www.lebigdata.fr/python-langage-definition>.
- [12] [En ligne]. Available: <https://www.journaledunet.fr/web-tech/dictionnaire-du-webmastering/1203355-framework/>.
- [13] [En ligne]. Available: [https://fr.wikipedia.org/wiki/Flask_\(framework\)](https://fr.wikipedia.org/wiki/Flask_(framework)).
- [14] [En ligne]. Available: <https://www.journaledunet.fr/web-tech/dictionnaire-du-webmastering/1203255-html-hypertext-markup-langage-definition-traduction/>.
- [15] [En ligne]. Available: <https://www.journaledunet.fr/web-tech/dictionnaire-du-webmastering/1203277-css-cascading-style-sheets-definition-traduction/>.
- [16] [En ligne]. Available: javascript <https://www.journaledunet.fr/web-tech/dictionnaire-du-webmastering/1203585-javascript/>.
- [17] [En ligne]. Available: <https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Objects/JSON>.
- [18] [En ligne]. Available: <https://research.aimultiple.com/data-augmentation/>.
- [19] [En ligne]. Available: <https://research.aimultiple.com/data-augmentation-techniques/>.
- [20] [En ligne]. Available: <https://github.com/rodrigopivi/Chatito>.



AITHOUNA Basma



BENBELLA Taha



FAKIL Mohammed

Chatbot Touristique

Université Hassan II Mohammedia-Casablanca

Faculté des Sciences et Techniques de Mohammedia

RÉSUMÉ :

Dans ce projet de fin d'étude, nous avons l'honneur de travailler au sein de notre faculté des science et Technique de Mohammedia sous la supervision de nos enseignants et surtout sous l'encadrement du Pr AYAD Habib qui nous a offert un excellent cadre de travail.

L'objectif de notre projet était la réalisation d'un chatbot touristique en anglais qui permettra au touriste et au voyageur de toutes nationalités de découvrir la ville de Marrakech et ses super endroits (Hotels, restaurants, monuments).

En appliquant tous les techniques et les informations requises pendant la formation LST IRM, on a pu aboutir notre objectif et réaliser un chatbot performant avec une interface artistique, simple et utile.

Nous étions très intéressés par ce domaine des chatbots et de l'intelligence artificiels que nous avons travaillée avec une grande motivation et esprit de groupe.

27/07/2021