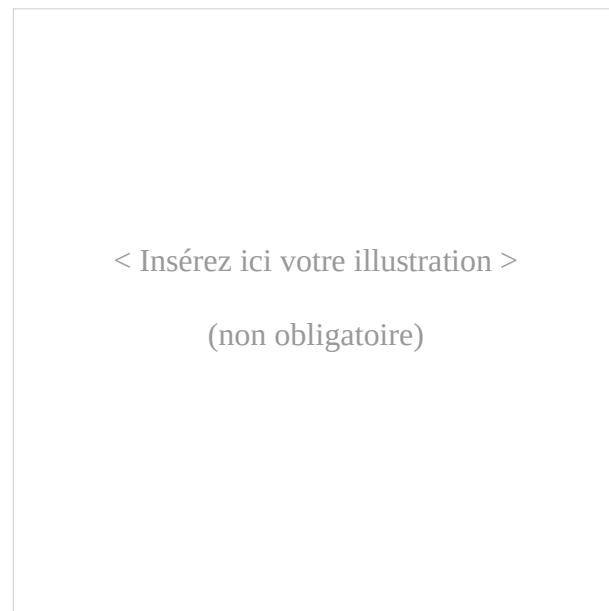


## < TITRE DE VOTRE MÉMOIRE DE PROJET DE SEMESTRE >



< Insérez ici votre illustration >

(non obligatoire)

Projet de semestre présenté par

< Prénom NOM >

**Informatique et systèmes de communication avec orientation en**  
< Nom complet de votre orientation >

< Mois, année >

Professeur-e HES responsable

Mandant (si existant)

< Prénom Nom >

< Prénom Nom >

Légende et source de l'illustration de couverture :

## TABLE DES MATIÈRES

<i>Remerciements (style « Titre 1 ») .....</i>	<i>vi</i>
<i>RÉsumÉ (style « Titre 1 ») .....</i>	<i>vii</i>
<i>Liste des acronymes.....</i>	<i>viii</i>
<i>Liste des illustrations.....</i>	<i>ix</i>
<i>Liste des tableaux.....</i>	<i>x</i>
<i>Liste des annexes.....</i>	<i>xi</i>
<i>Introduction .....</i>	<i>1</i>
<b>1. Chapitre 1 : Définition de Chatbot.....</b>	<b>4</b>
1.1.    Un Chatbot, qu’ est-ce ?.....	4
1.2.    Trois composants d’ un Chatbot .....	5
a)    L’ interface.....	5
b)    Le moteur de traitement de données .....	5
c)    Le moteur de réponses .....	6
1.3.    trois types de Chatbots.....	6
a)    Les Chatbots de Menus.....	6
b)    Les Chatbots de règles.....	9
c)    Les Chatbots “intelligents” .....	11
<b>2. Chapitre 2 : Étude des solutions existantes.....</b>	<b>16</b>
2.1.    Agenda.ch.....	16
2.2.    Klara.ch.....	17
2.3.    Meetme.io .....	18
2.4.    Comparaison à ce projet .....	18
<b>3. Chapitre 3 : Detail technologique.....</b>	<b>20</b>
3.1.    Natural Language Processing .....	20
a)    Natural Language Understanding.....	24
b)    Natural Language Generation .....	25
3.2.    TAPAS.....	25
a)    BERT .....	26
b)    TAPAS : Fonctionnement .....	26
3.3.    Rasa .....	26
3.4.    Rasa Pro.....	26
<i>Conclusion (style « Titre 1 »).....</i>	<i>27</i>
<i>Annexes (style « Titre 1 ») .....</i>	<i>28</i>
Annexe 1.....	29
Annexe 2.....	30

<b>Annexe 3.....</b>	<b>31</b>
<b>RÉfÉrences documentaires (style « Titre 1 ») .....</b>	<b>32</b>

*NB : La table des matières est à actualiser par le menu contextuel « Mettre à jour l'index ».*

Nom, Prénom – Titre abrégé – Projet de semestre – Mois, Année

< Insérez ici votre dédicace > (facultatif)

## REMERCIEMENTS (STYLE « TITRE 1 »)

< Formulez ici vos remerciements aux personnes qui vous ont aidé dans la réalisation de votre travail. Les remerciements sont rédigés dans le style « Corps de texte » >.

*NB : les Titres « Titre 1 non numérotés » sont basés sur le titre 1 mais sont en dehors de la numérotation des chapitres. Ils apparaissent néanmoins dans la table des matières*

## RÉSUMÉ (STYLE « TITRE 1 »)

< Insérez ici votre illustration >

(obligatoire)

### Candidat-e :

< NOM PRENOM >

Filière d'études : ISC

Professeur-e(s) responsable(s) :

< NOM PRENOM >

**En collaboration avec :** < Nom de l'entreprise >

Travail de semestre soumis à une convention de stage en entreprise : < oui/non >

Travail soumis à un contrat de confidentialité : <oui/non>

*Attention : Tout le résumé doit tenir sur une seule page*

## **LISTE DES ACRONYMES**

**NLP** Natural Language Processing

**NLU** Natural Language Understanding

**NLG** Natural Language Generation

**I.A.** Intelligence Artificielle

**UX** User Experience

**POS** Part-of-Speech

**NER** Named Entity Recognition

## LISTE DES ILLUSTRATIONS

Illustration 1: Graphique de la relation entre les trois composants de base .....	5
Illustration 2: Graphique d'un Chatbot de Menus .....	7
Illustration 3: Arbre de décision d'un Chatbot de Menus.....	8
Illustration 4: Graphique d'un Chatbot de règles .....	9
Illustration 5: Exemple de sélection de règle pour Chatbot par règles .....	10
Illustration 6: Graphique d'un Chatbot Intelligent.....	12
Illustration 7: Pyramide de la difficulté technologique et UX selon le type de Chatbot .....	15
Illustration 8: Séquence de traitement NLP .....	20
Illustration 9: Représentation de la relation NLP, NLU et NLG.....	24

### Références des URL

- URL01            <https://www.ovationc xm.com/blog/3-kinds-chatbots-youll-meet>
- URL02            <https://www.geeksforgeeks.org/natural-language-processing-nlp-tutorial/>
- URL03            <https://datasolut.com/natural-language-processing-vs-nlu-vs-nlg-unterschiede-funktionen-und-beispiele/>
- URL04            <https://commons.wikimedia.org/w/index.php?curid=906980>
- URL06            <ce-site.ch/monrapportdestage.pdf>

## LISTE DES TABLEAUX

Tableau 1: Lot de données n°1.....**Erreur ! Signet non défini.**  
Tableau 2: Lot de données n°2.....**Erreur ! Signet non défini.**

*N.B. Si vous avez peu de tableaux, vous pouvez les intégrer à la table des illustrations.*

### Références des URL

URL02 ce-site.ch/bli/bla/blo/blou

URL05 ce-site.ch/publications/documents/ rapports/rapportsdestage/  
monrapportdestage.pdf

## **LISTE DES ANNEXES**

Annexe 1 .....	29
Annexe 2 .....	30
Annexe 3 .....	31



## INTRODUCTION

De nos jours et comme il a été le cas par le passé, que ce soit pour aller au médecin ou au restaurant ; Il est commun de placer des réservations. Or pour ce faire il est d'usage de passer un coup de fil ou dans d'autres cas de placer la réservation en se rendant au dit lieu du rendez-vous. Aujourd'hui, il est possible de placer ses réservations directement depuis de nombreux sites web. Qu'ils soient possédés par le particulier chez qui on souhaite placer rendez-vous ou par une autre entité chargée de gérer l'aspect de placement de réservations et/ou rendez-vous. Ces sites web permettant de prendre rendez-vous se présentent très régulièrement sous la forme de boutons cliquables et de quelques champs à remplir telle que la date du rendez-vous, le nom/prénom et autres informations. Cependant, il est bien plus rare qu'une alternative à cela existe et ne nécessitant pas de devoir se rendre sur une page internet quelconque. Le projet qui sera réalisé a pour optique d'apporter un autre moyen de placer une réservation se trouvant à l'intersection d'un appel téléphonique et un simple formulaire cliquable : Un Chatbot servant à prendre des rendez-vous depuis des applications de messageries telles que What's app ou Telegram ou alors pouvant être aisément inclu dans un site web existant et qui permettra de rendre plus fluide le processus de prise de rendez-vous.

Ce Chatbot s'apparente plus au premier abord à un composant applicatif auquel il est possible de fournir du texte et en recevoir en sortie. Bien que le but premier soit de fournir un Chatbot avec lequel il sera possible de converser sur une application de messagerie, il est tout à fait possible de mettre en place une section Chatbot sur un site quelconque et d'y lier le Chatbot, le rendant de ce fait facile à mettre en place sans avoir à se soucier de l'interface graphique de l'unique nécessité du Chatbot étant un champ textuel mis à disposition à l'utilisateur.

Une des raisons primaires de ma décision de réaliser ce projet étant qu'à présent, en 2024, de nombreux progrès ont été réalisés dans le secteur du Machine Learning et de l'I.A. et bien que de nombreuses compagnies ont commencé à doucement mais sûrement à implémenter des technologies diverses en termes d'I.A. ou Machine Learning, il y a encore un grand nombre de secteurs dans lesquels la présence de ces dernières se fait encore discrète ou simplement car peu ont considéré faire usage de ces technologies dans ces derniers. Le secteur en question présentement est celui du Service Client et plus spécifiquement l'assistance à réservation et prise de rendez-vous. Je trouve tout particulièrement pertinent l'exploration de cette voie qu'est l'usage de ces technologies pour la réalisation d'un Chatbot pour prendre rendez-vous car moi-même devant de temps à autre prendre rendez-vous, il est bien difficile pour bon nombre de gens de passer des coups de fil, moi inclus. C'est pour cela que fournir une solution qui soit modulaire et ne nécessitant par d'architecture particulière que simplement un moyen d'incorporer une boîte de dialogue ou champ textuel quelque soit la plateforme choisie faciliterait grandement son utilisation par les particuliers de tout secteurs et qui fournirait une alternative viable aux appels et sites web.

Ce rapport précède le travail de Bachelor Informatique et Systèmes de Communications à L'Haute Ecole du Paysage, d'Ingénierie et d'Architecture de Genève et a pour objectif d'expliciter les divers concepts proéminents dans le projet, l'étude des divers technologies existantes pouvant servir à la réalisation de ce projet, l'explicitation des choix technologiques découlant de cette étude, de montrer des exemples concrets de leur utilisation à l'aide de quelques prototypes et de finalement étaler le plan de conception du travail allant être réalisé.

Actuellement, la réalisation de ce rapport s'est effectuée de manière régulière chaque semaines lorsque le temps me le permettait et un état des lieux fut régulièrement effectué à l'aide d'entretiens hebdomadaires avec M. Niklaus Eggenberg. Afin de réaliser ce rapport, l'outil Git a servi à versionner et stocker les divers prototypes réalisés pour tester les

technologies ainsi que le journal de bord tenu à jour régulièrement pour faire compte du travail effectué à chaque session de travail. Le stockage s'est effectué sur la plateforme Gitlab mise à disposition par la HES et sur un projet Git possédé par M. Niklaus Eggenberg. L'outil Google Docs fut d'une grande aide lors de l'écriture d'ébauches, de prises de notes, d'explications préemptives des divers technologies recherchées et pour la formulation de mes idées concernant le travail à réaliser par la suite.

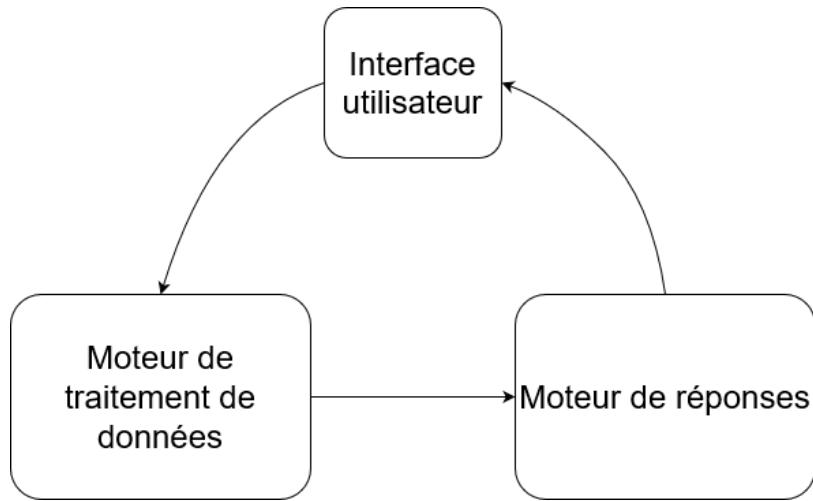
Ce rapport va par conséquent servir de fondation à la réalisation prochaine du Chatbot. Concernant les divers section abordées dans ce rapport, elles se comptent au nombre de sept chapitres **À VOIR EN FONCTION DE L'EVOLUTION DE LA REDACTION (CE QUI SUIT N'EST PAS FINAL)**. Le premier définira de manière concrète et conceptuelle ce qu'est un Chatbot. Le deuxième portera sur l'étude des solutions existantes de réservation et prises de rendez-vous se trouvant sur le marché et comment elles comparent à celle proposée ici. Le troisième sera une énonciation et explicitation des divers concepts piliers composant les divers structures de Chatbot énoncées au chapitre précédent. Le quatrième a pour but de faire un état des nombreuses technologies existantes et utiles pour la réalisation de ce projet. Le cinquième chapitre me servira à présenter les quelques prototypes que j'ai pu réaliser de certaines des technologies énoncées précédemment ainsi que mes réflexions les concernant. Le sixième a pour simple but de définir dans quelle direction le projet ira structurellement parlant et quelles technologies seront usées pour sa réalisation. Et finalement, le septième élaborera dans le détail toute la partie de planification et conceptualisation du projet.

## 1. CHAPITRE 1 : DEFINITION DE CHATBOT

Avant même de commencer à parler technologies, il est important d'expliciter concrètement ce qu'est un Chatbot.

### 1.1. UN CHATBOT, QU'EST-CE ?

Les Chatbots sont des applications disponibles sous une multitude de formes et plateformes et ayant pour but premier d'engager dans une conversation avec un utilisateur afin de tenter au mieux de répondre à quelque requête que ce soit. Les Chatbots font affaire à un grand nombre de requêtes variées comme fournir des renseignements, fournir un service ne nécessitant pas d'opérateur humain, faire office de service client afin de tenter de résoudre tout éventuel problème dans la limite des capacités du Chatbot (Le cas échouant, le bot peut rediriger l'utilisateur vers un opérateur humain), récolter du feedback, et bien d'autres usages encore. Bien que des Chatbots existent dans un nombre incalculable de formes et structures, il y a néanmoins une manière de décomposer n'importe quel bot en une composition de trois éléments distincts



*Illustration 1: Graphique de la relation entre les trois composants de base*

*Source : Rodrigues dos Santos Fabio*

## 1.2. TROIS COMPOSANTS D’UN CHATBOT

### a) L’INTERFACE

Aucun Chatbot ne pourrait avoir la dénomination de *Chatbot* s’il ne possédait pas d’interface avec laquelle l’utilisateur peut interagir. Au plus simple, une interface pourrait être une simple ligne de commande de terminal et au plus poussé, une fenêtre de dialogue ayant l’apparence d’un logiciel de messagerie comme sur nos smartphones. L’essentiel ici est que l’interface mise à disposition comporte au moins une zone de saisie textuelle ou autre support de communication tel que des boutons cliquables par exemple. C’est au travers de ce support que l’utilisateur fera parvenir ses diverses requêtes au bot pour traitement ultérieur.

### b) LE MOTEUR DE TRAITEMENT DE DONNÉES

Une fois que l’utilisateur a exprimé sa demande au travers d’un des canaux de communications disponibles, cette demande sous forme de texte va être immédiatement acheminée au moteur de traitement de donnée. Les requêtes utilisateurs dans le cas où le

canal de communication est un champs de texte se présentent sous la forme de suites de caractères. Ce qui est tout à fait compréhensible pour un humain mais pas pour une machine, c'est donc pourquoi un traitement se doit d'être appliqué au texte utilisateur. Le texte se présentant sous formes de phrases sera verra le plus souvent mis sous la forme de mots-clés ou instructions pouvant eux être compris par la machine.

### c) LE MOTEUR DE RÉPONSES

Une des dernières étapes du cycle d'échange entre l'utilisateur et le bot est le passage des données précédemment traitées dans le moteur de réponse. Son rôle est de faire sens des divers mots-clés et/ou instructions qu'il a reçues et d'envoyer à l'utilisateur la réponse correspondant le plus à sa requête.

Ces trois composants, une fois mis ensemble forment un Chatbot fonctionnel capable de recevoir des requêtes d'utilisateurs, de les rendre compréhensibles par le système et d'en sortir la réponse adéquate à l'utilisateur et ce ainsi de suite jusqu'à ce que l'échange prenne fin.

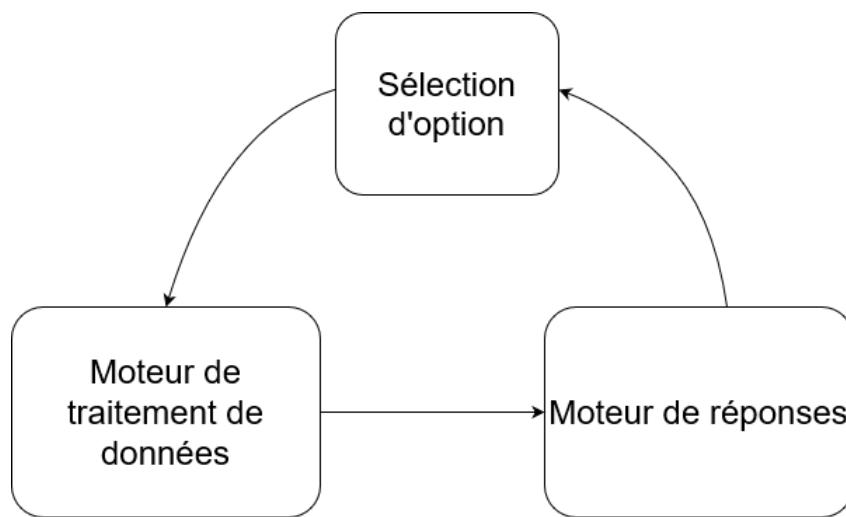
## 1.3. TROIS TYPES DE CHATBOTS

Ce simple cycle d'échange de données expliqué ci-dessus permet à présent de diviser cette fois les Chatbots en trois catégories distinctes étant les variantes de Chatbots les plus communes.

### a) LES CHATBOTS DE MENUS

Une des formes les plus basiques de Chatbots qui existe, les Chatbots fonctionnant à l'aide d'un menu.

Les Chatbots de Menus, bien qu'étant factuellement très simplistes ne sont pour autant pas mis de côté du fait de leur aspect simpliste et parfois préféré par rapport à d'autres types de Chatbots. En effet, un Chatbot de Menus permet d'éviter tout soucis qui pourraient advenir lors de l'implémentation des autres types de Chatbots plus bas. Grâce à leur canal de communication on ne peut plus facile à utiliser pour l'utilisateur : Des boutons. L'intégralité de l'échange est réalisée grâce aux pressions successives des multiples boutons s'affichant à l'écran de l'utilisateur et ne nécessitant aucune autre forme d'interaction de ce dernier.

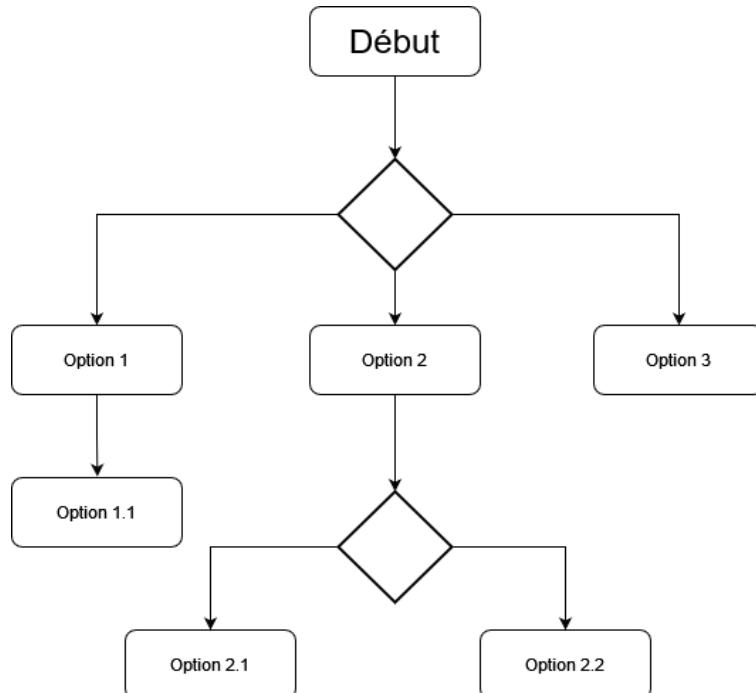


*Illustration 2: Graphique d'un Chatbot de Menus*

*Source : Rodrigues dos Santos Fabio*

Ce type de bots est notamment très prisé dans des scénarios où les requêtes pouvant être effectuées par l'utilisateur sont prédéfinies comme pour un système de commande de plats en ligne, un système de support qui permet d'imiter un système similaire existant déjà au format téléphonique étant la demande à l'utilisateur de presser des boutons sur leur téléphone pour spécifier quel type d'aide ce dernier nécessite; ici représenté sous la forme de boutons et qui permet ensuite de donner des informations à l'opérateur humain avant même qu'il n'ait à demander quel type de soucis le client rencontre ou encore lors de placement de réservations.

La séquence de pression de boutons peut être représentée sous la forme d'un arbre de décision. Comme le moyen d'interaction usuel de ce bot se représente sous la forme de boutons cliquables, le système impose un cadre prédéfini ainsi qu'un nombre limité d'interactions possibles et prohibe toute sortie de ce cadre. En effet, à chaque pression de bouton, l'utilisateur prends un des nombreux chemins existants dans cet arbre de décision jusqu'à arriver (ou non) à une réponse convenable et renvoyé à l'utilisateur au travers du moteur de réponses.



*Illustration 3: Arbre de décision d'un Chatbot de Menus*

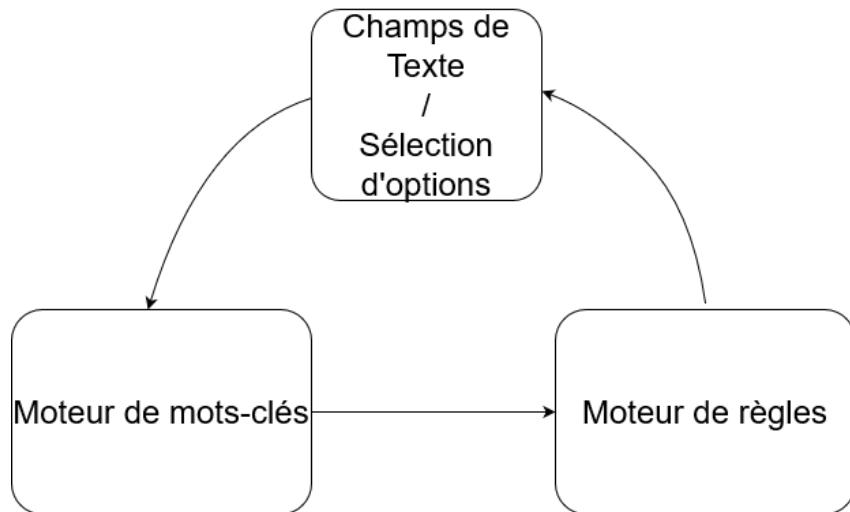
*Source : Rodrigues dos Santos Fabio*

Un des inconvénients possibles de ce Bot est en termes de scalabilité. Plus l'on souhaite étendre le panel de décisions possibles plus il faudra adapter l'affichage en conséquence et entre autres, un panel trop large pourrait être un désavantage plus qu'un avantage car bien que donner le plus de choix que possible à l'utilisateur semble être optimal, en avoir trop et donc obliger l'utilisateur à cliquer sur une trop grande quantité de boutons peut ruiner l'expérience utilisateur.

Un autre inconvénient possible est que bien que dans certains cas, il est davantage souhaité d'avoir un cadre fixe, il y en a d'autant plus d'autres où l'utilisateur souhaite effectuer une requête qui ne correspond à aucune des requêtes existantes ce qui obligerait soit à créer une très grande quantité de variations dans l'arbre de décision et qui n'arrangerait pas plus le problème qu'il ne l'aggraverait.

b) **LES CHATBOTS DE RÈGLES**

Se trouvant un cran au-dessus niveau complexité, se trouvent les Chatbots de Règles. Ces bots ajoutent un degré de liberté qu'il n'est pas aussi aisément trouvable avec un Chatbot de menus grâce à l'utilisation de règles.



*Illustration 4: Graphique d'un Chatbot de règles*

*Source : Rodrigues dos Santos Fabio*

Ce type de bot est presque aussi populaire que ceux à base de menus car tout comme eux, ils répondent à des demandes similaires. Les interfaces souvent rencontrées avec ce genre de Chatbots sont soit des boutons ou cette fois-ci des champs de texte. Il est important de noter la distinction entre les Chatbots de menus et de règles, car l'un possède un chemin prédéfini avec des options se succédant, alors que l'autre possède un ensemble de règles qui selon les attributs ressortis de la requête utilisateur, la règle la plus adéquate sera sélectionnée.

Un exemple pour le cas avec boutons serait qu'au lieu qu'un bouton ne fasse simplement avancer dans l'arbre de décision, que chaque boutons soit pondéré et que même avec une succession de pressions de boutons différentes on puisse arriver à la même réponse.

Dans le cas où l'entrée utilisateur est un texte, la méthode la plus facile et répandue est le simple parage de mots-clés dits « Tokens ». Selon les tokens ayant été ressortis de la requête, une règle comprenant un certain nombre de tokens correspondant ou tous se verra sélectionnée.

Ceci est une phrase **d'exemple**, bien qu'elle paraisse importante



Illustration 5: Exemple de sélection de règle pour Chatbot par règles

Source : Rodrigues dos Santos Fabio

Il existe deux approches possibles à la tokenisation du texte

La première consistant à faire usage d'un dictionnaire de mots-clés établi au préalable. Lorsqu'une entrée utilisateur se voit traitée par le moteur de mots-clés, ce dernier ira simplement vérifier si le token est présent dans le dictionnaire. Si c'est le cas, alors il est ajouté à la suite de tokens qui sera envoyée au moteur de réponse.

La deuxième, elle, consiste à faire usage du NLP (Natural Language Processing) qui permet de faciliter davantage l'implémentation du dictionnaire. Car dans la première approche, une limitation qui pourrait rapidement se faire ressentir est pour commencer le fait que pour chaque tokens ajoutés dans le dictionnaire, il faudra les associer à des règles et donc plus il y a de tokens, davantage il faudra créer d'associations règles-tokens. De plus, pour chaque token il peut exister une infinité de variations à cause de fautes

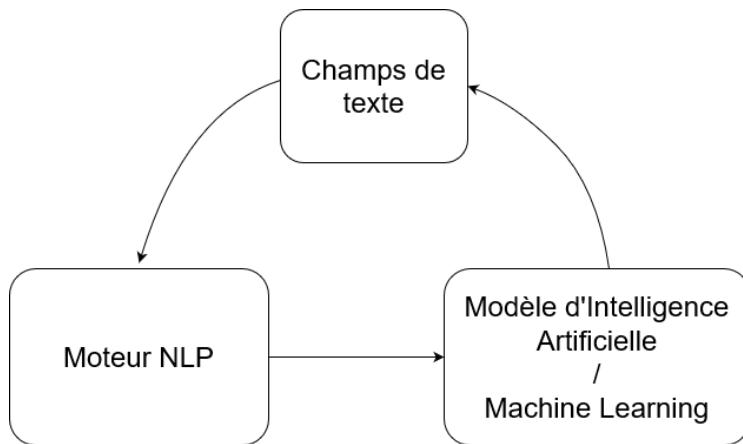
d'orthographe ou tokens similaires en écriture et/ou sens ce qu'une simple comparaison mot à mot ne pourrait pas détecter correctement.

En d'autres termes, le problème sous-jacent à une implémentation si simpliste est un de scalabilité du projet. Pour un petit projet, ce n'est pas un grand soucis mais avec le temps cela pourrait vite devenir ingérable. C'est donc pour cette raison qu'il est judicieux de faire usage d'un moteur NLP si le projet commence à prendre de l'ampleur pour que la première approche soit viable car ce dernier permet d'éviter de devoir par exemple y inscrire toutes les variations d'un mot grâce à l'usage de divers techniques de NLP voire de regrouper des familles de mots sous un seul type de token. De ce fait, la taille du dictionnaire final se verra grandement réduite et donc même avec un nombre de règles, l'application reste bien plus maintenable et scalable.

Les inconvénients à prendre en compte venant avec l'utilisation d'un Chatbot de règles est que bien qu'il permette un plus grand degré de liberté avec l'usage de champs textes et par conséquent de tokens qui élargissent le panel de possibilités à disposition de l'utilisateur en termes de requête, on reste tout de même dans un cadre restreint car tous les comportements sortant du moteur de règles doivent être implémentés au préalable et continuellement mis à jour au besoin. Ce qui donne l'illusion à l'utilisateur qu'il peut poser quelque requête que ce soit mais il se verra vite confronté à une quantité de réponses limitées s'il sort trop du cadre initial par inadvertance.

### c) LES CHATBOTS “INTELLIGENTS”

Le dernier type de Chatbot étant à la fois le plus complexe et pouvant apporter le plus de qualité en termes d'expérience utilisateur sont les Chatbots dit « Intelligents ».



*Illustration 6: Graphique d'un Chatbot Intelligent*

*Source : Rodrigues dos Santos Fabio*

Les Chatbots intelligents permettent de pallier le problème commun aux deux types de Chatbots précédent étant le contexte restreint. Dans le cas du Chatbot de règles, il a bien été soulevé qu'ajouter de nouveaux tokens et règles rends le problème exponentiellement complexe dû à l'infinité des tournures que peut prendre une phrase. Or, un Chatbot intelligent est capable d'outrepasser ces limitations par sa capacité à comprendre le sens des mots, le contexte dans lequel ils sont employés et d'engager dans une conversation avec un utilisateur en se rappelant des informations que ce dernier a pu fournir tout au long de la discussion. Étant donné la grande variété de modèles d'intelligence artificielle, il est difficile de donner un exemple d'implémentation spécifique. Le point commun entre tous cependant est qu'ils ont tous pour but de simuler une discussion fluide et dynamique avec l'utilisateur, comme s'ils discutaient avec un agent humain.

Ce type de Chatbot est souvent retrouvé actuellement dans des assistants virtuels tels qu'Alexa, Google Assistant, Siri, Amazon Echo et bien d'autres. Il en existe aussi sous la forme de Chatbots web comme le très populaire ChatGPT, Google Bard, Bing AI qui lui est une intelligence artificielle de type Générative qui a pour but de générer du nouveau contenu à partir d'un contenu existant. Ce qui, bien que cela est intéressant, n'est pas le focus du projet ici.

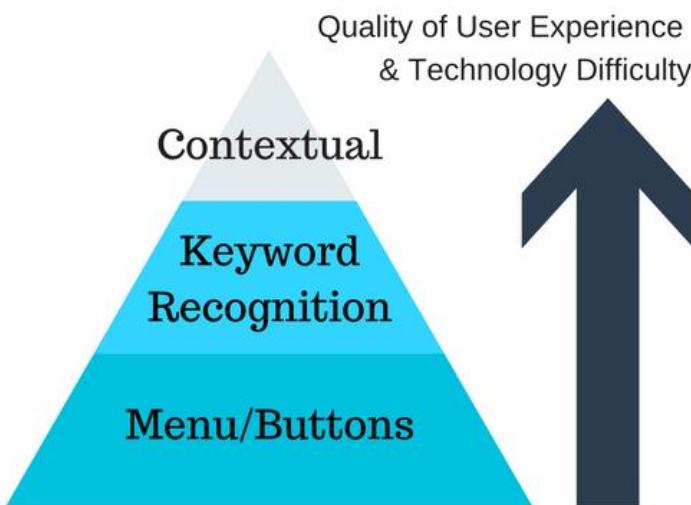
Cependant, un aspect partagé par toutes ces I.A. est que dans tous leurs traitements de requêtes utilisateurs est l'usage de NLU (Natural Language Understanding). Selon le graphique présenté plus haut, une fois que l'application reçoit la requête utilisateur sous forme de texte, elle subit généralement un prétraitement grâce à l'utilisation d'un moteur NLP. Ce prétraitement peut être plus ou moins utile selon le modèle d'I.A. ou de Machine Learning utilisé car certains modèles comme des I.A. conversationnelles sont programmée pour se charger de toute la partie de Tokenisation et autres traitements NLP avant d'appliquer un traitement NLU afin de comprendre le contexte, intention et sens des phrases et mots présents dans la requête. Le tout étant finalement traité par le modèle correspondant et sa réponse renvoyée à l'utilisateur.

Il est évident qu'un Chatbot de ce type apporte une grande plus-value à l'expérience utilisateur car ce dernier donne l'impression de comprendre quelque requête que ce soit et de rendre l'expérience bien plus personnelle qu'un Chatbot avec règles ou menus pourrait fournir. Hélas, cela ne s'obtient pas aussi aisément car tout comme le moteur NLP qui nécessite une certaine quantité de données pour être entraîné et efficace, il faut une vaste quantité de données d'entraînement si l'on souhaite avoir un Chatbot capable de gérer toute situation se présentant ou alors une quantité assez large et variée pour au moins gérer la plupart des situations dans le cadre d'un projet de moins grande envergure. Cela implique qu'il n'est pas impossible de mettre en place un Chatbot intelligent pour un petit projet mais qu'il faut nécessairement assez de données à disposition. C'est pourquoi un prétraitement par un moteur NLP contribue à nécessiter le moins de données que possible si l'entrée utilisateur peut être transformée en quelque chose de plus général et moins propice à être des cas uniques non pris en compte.

L'essentiel à retirer de ces types de Chatbots se présente ainsi :

- Les Chatbots de Menus sont simplistes tant niveau interface que dans la manière de les utiliser. Ils permettent facilement d'arriver à une réponse en peu d'entrées utilisateur, la complexité d'implémentation étant basse les rend attractif selon le besoin, surtout si le cadre dans lequel il est utilisé est déjà un cadre restreint. Cependant, l'expérience n'est pas très personnelle et ne permet pas de sortir du cadre imposé par le menu lorsque la réponse à la requête utilisateur pourrait ne pas se trouver dans le cadre imposé par l'application. Très utilisé quand l'on sait d'avance que l'utilisateur ne peut pas sortir du cadre imposé et que l'on souhaite avoir un Chatbot nécessitant le moins de maintenance possible.
- Les Chatbots de règles permettent de réaliser des exploits similaires à ceux d'un Chatbot de Menus mais en permettant une plus grande liberté côté utilisateur avec la possibilité de mettre un champs de texte à sa disposition. En plus d'une liberté accrue, l'expérience fournie sera plus personnelle car l'utilisateur peut formuler par ses propres mots sa requête. Or, un soucis de cadre existant persiste car ce qui définit le cadre est la quantité de règles et mots-clés gérés par le Chatbot. S'il n'y a pas assez de règles ou mots-clés pour couvrir tous les cas d'usage, cela empêterait sur l'expérience utilisateur et faire en sorte qu'aucune solution viable ne soit trouvée.
- Les Chatbots intelligents sont très prisés par beaucoup en ce moment de par leur avantages attractifs étant la personnalisation de chaque discussions, de la liberté donnée à l'utilisateur en matière de requêtes tout en gardant un cadre et que le Chatbot tentera d'y ramener l'utilisateur s'il s'en écarte trop lors d'un échange, la facilité à implémenter un système dynamique sans avoir à se préoccuper de tout l'aspect de compréhension par NLU et traitement du texte par NLP et de n'avoir qu'à donner la sortie de ces traitements au modèle pré-entraîné qui trouvera la réponse adéquate à la requête utilisateur. Mais pour y

parvenir le plus grand obstacle reste toujours les données utilisées lors de l'entraînement des divers composants du Chatbot qui impactent grandement sa qualité. S'il n'y en a pas assez ou qu'elles ne soient pas assez variées pour coller aux spécificités souhaitées, le bot pourrait peiner en premier lieu à comprendre l'entrée utilisateur et par la suite à trouver la solution adéquate. De plus, l'usage de modèles d'I.A. ou de Machine Learning nécessitent un plus grand niveau de compréhension de ces derniers afin de déterminer quels modèles et implémentations sont les plus adéquates aux cas d'usages du projet.



*Illustration 7: Pyramide de la difficulté technologique et UX selon le type de Chatbot*

*Source : Ovationcsm, ref. : URL01*

L'illustration ci-dessus démontre de manière visuelle la relation entre la difficulté de la technologie elle-même dans la création de Chatbots par Menus, par Règles ou Intelligents. Il est important de noter que le choix final du type de Chatbot souhaité ne dépend pas uniquement du niveau de qualité de l'expérience utilisateur fournie car sinon seul les Chatbots intelligents seraient utilisés mais la complexité d'implémentation, l'étendue des possibles interactions (tant larges que plus restreintes), ressources à disposition et autres paramètres sont à prendre en compte afin de choisir le Chatbot répondant au mieux aux attentes du projet.

## 2. CHAPITRE 2 : ÉTUDE DES SOLUTIONS EXISTANTES

### 2.1. AGENDA.CH

Agenda.ch est une application de réservation de rendez-vous Suisse basée à Genève et lancée en 2011. Agenda.ch présente actuellement 5 services dédiés et 1 personnalisable à la demande. Les services à disposition sont :

1. Application dédiée aux physiothérapeutes
2. Application dédiée aux thérapeutes et autres services médicaux
3. Application dédiée aux centres de beauté et Bien-être
4. Application dédiée aux centres sportifs et loisirs
5. Application à visée d'administration publique

Fondamentalement, Agenda.ch fourni une application qui peut s'apparenter à un Chatbot de Menus. Peu importe la forme, elle permet de réserver des ressources étant une salle de sport, une heure pour un rendez-vous médical, une séance pour une coupe de cheveux, etc., de choisir son créneau horaire et finalement tout autre information complémentaire.

L'aspect Chatbot n'est pas le seul utilitaire fourni par cette entreprise :

- L'application n'est pas fournie sous forme d'un module à intégrer ou d'une API à appeler mais elle est disponible chez les serveurs loués par Agenda.ch et tout l'aspect de réservation est mis en place chez Agenda.ch et non chez le client.
- Selon le type de service fourni, il est possible de stocker une multitude de fichiers pertinente à la réservation des clients sur les serveurs d'Agenda.ch
- Selon le type de service fourni, un grand panel de fonctionnalités sont disponibles.

- La possibilité de configurer des messages envoyés automatiquement aux clients pour des rappels de réservations, informations, newsletter, etc.

Ce qui ressort de l'analyse de l'application d'Agenda.ch est qu'elle ne demande aucune implémentation par le client lui-même et fourni l'ensemble de ses services depuis leur site web que ce soit la configuration ou l'utilisation même du système de réservations. L'application est donc détachée du site web du client ce qui peut être vu comme un avantage étant donné que cela facilite son usage car il ne suffit que d'intégrer un simple bouton « réserver » sur le site.

Si l'on se penche sur les tarifs, on remarque qu'il y en a trois. Un à 35 CHF/mois qui n'offre pratiquement que le système de calendrier et de rappel aux clients d'une réservation. Ce n'est qu'à partir de 60 CHF/mois que l'on obtient les fonctionnalités telles qu'un mini-site permettant aux clients d'effectuer leurs réservations dessus ainsi que d'avantages de fonctionnalités en ligne pour faciliter l'aspect réservation. Le dernier tarif étant à 80 CHF/mois propose principalement des systèmes de gestion de documents et ressources diverses directement effectuées par Agenda.ch, pratique si l'on ne possède pas de grande infrastructure ou que l'on ne souhaite pas avoir à se préoccuper d'en gérer une soi-même. Il existe également des modules supplémentaires ou agendas supplémentaires selon des coûts additionnels variables.

## 2.2. KLARA.CH

Klara.ch est une entreprise Suisse fondée en 2016 basée à Berne et a pour but de fournir un panel varié d'outils d'administration. Contrairement à la solution précédente, Klara.ch ne se spécialise pas uniquement dans la mise en place d'un système de réservations mais d'une panoplie d'outils d'administration comme : Des gestionnaires de client, gestionnaires de budget, gestionnaires d'inventaire, création de shop en ligne, mise en place d'un système de réservations et bien d'autres. L'application de réservation en ligne ainsi que les autres fonctionnent sous forme de widgets soit des modules applicatifs que le

client intègre directement à son site web. La configuration de ce dernier se fait directement au travers du panneau de configuration Klara. Le système de réservation se présente sous la forme d'un Chatbot de Menus, similaire à celui d'Agenda.ch. Le tarif pour le widget de réservation en ligne est de 39 CHF/mois.

La solution de Klara.ch semble intéressante de par son coût réduit par rapport à Agenda.ch et permet d'avoir l'ensemble des fonctionnalités disponibles sur le site client et ne nécessitant pas de devoir naviguer sur une autre page internet. Cependant le degré de customisation et de fonctionnalités supplémentaires spécifiques à certains services ne semble pas comparable à celui d'Agenda.ch, expliquant ainsi son coût plus élevé.

### **2.3. MEETME.IO**

Meetme.io est une application de réservation de rendez-vous Suisse basée à Lausanne et ayant pour objectif de fournir un service de réservation en ligne et de gestion d'agenda grâce à une application ou depuis un panneau d'informations. Meetme.io possède un catalogue de fonctionnalités bien plus réduit comparé aux solutions précédentes ne fournissant uniquement un système de formulaire similaire à celui d'Agenda.ch dans le sens où il n'est accessible uniquement par un lien qui amène le client sur le site de meetme.io à l'url du formulaire de réservation du service souhaité et un système de rappel par sms. Similairement aux solutions précédentes, le formulaire de Meetme est comparable à un Chatbot de Menus. Le tarif pour avoir une quantité raisonnable réservation disponibles s'élève à 69 CHF/mois.

### **2.4. COMPARAISON A CE PROJET**

Après analyse de quelques solutions Suisses disponibles sur le marché actuellement, il est clair qu'il y a une tendance à réaliser un système de réservation par internet. Tous partagent le point commun que le système fonctionne sous la forme d'un formulaire étant

ici l'équivalent d'un Chatbot de Menus. Bien que certaines solutions offrent parfois plus de fonctionnalités que simplement un système de réservation ou proposent des systèmes de réservation avec des fonctionnalités étendues et spécifiques à certains secteurs, aucun ne propose de solution par téléphone uniquement. C'est donc ici que va se placer la solution que je vais proposer étant donné qu'elle permettra de placer des réservations depuis un simple échange par texte avec un Chatbot par téléphone et d'ensuite éventuellement de récupérer facilement l'ensemble des réservation sous format Caldav ou d'exporter le tout en tant que calendrier google.

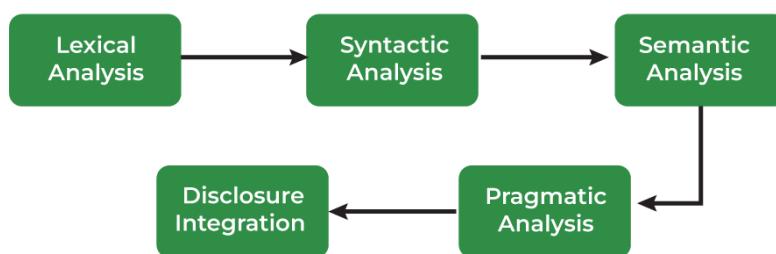
### 3. CHAPITRE 3 : DETAIL TECHNOLOGIQUE

Au cours de mes recherches concernant les diverses technologies associées à la réalisation de mon Chatbot, je me suis retrouvé à étudier quelques concepts jusque là nouveaux pour moi et qui ont par la suite servi à affirmer mon choix technologique pour ce projet.

#### 3.1. NATURAL LANGUAGE PROCESSING

La première technologie et une des plus importantes dans le cadre de ce projet est celle du **Natural Language Processing** (Traitement du langage naturel) (NLP). Cette technologie est une pierre angulaire au sein du domaine du machine learning et de l'intelligence artificielle. En effet, le principal intérêt de ce dernier est de permettre de prendre quelconque texte et d'appliquer divers traitements dessus en identifiant des patrons dans les phrases et mots et qui donne en sortie une suite de données structurée dont les divers modèles seront capables de les exploiter afin qu'eux aussi appliquent des traitements variés sur ces données pour finalement obtenir une réponse. Ce qui sans le NLP serait infiniment plus ardu.

L'être humain en quelque sorte applique une multitude de techniques de NLP tous les jours pour faire sens des paroles et textes que nous lisons. De manière générale, tout traitement NLP suit globalement cette séquence :



*Illustration 8: Séquence de traitement NLP*

*Source : geeksforgeeks.org, ref. : URL02*

Pour chaque étapes il existe une grande quantité de traitements divers, cependant je ne vais ici qu'en présenter une liste non exhaustive :

## 1. Analyse Lexicale

### a. Tokenisation

La Tokenisation est une des premières techniques NLP qui va se voir être appliquée à un texte donné. Cette dernière a pour but de séparer tout ce qui s'apparente à des mots sous forme de Tokens.

Un exemple de Tokenisation serait « Je vais marcher dehors » qui deviendrait « ‘Je’, ‘vais’, ‘marcher’, ‘dehors’ ».

### b. Lemmatisation et Stemming

La Lemmatisation et le Stemming sont présentés ensemble car ils ont un but très semblables étant de prendre un mot et de le réduire à une forme commune.

Pour se faire, la Lemmatisation va faire usage d'un dictionnaire de mots comprenant des groupes de mots similaires et étant tous lié par une même racine commune.

Par exemple : « Jouer, Joueur, Jeu, joué, ... » vont avoir pour mot racine « Jeu »

Ensuite, le Stemming lui, de par la signification du mot en anglais Stem voulant dire tige d'une plante et étant la base de cette dernière, consiste à prendre chaque mot et de les séparer en deux parties. Comme des mots de la même famille possèdent une base commune, retirer l'excès de lettres faisant permets d'en dégager une racine commune.

Par exemple : « Concourût, Concourir, Concourant, Concours » vont se voir réduits à « Concour ».

## 2. Analyse Syntaxique

### a. *Part-of-speech Tagging (POS)*

Le POS Tagging permet de donner un tag/label à chaque mots dans un texte afin de lui assigner sa classe correspondante grammaticalement parlant. L'intérêt principal de ce traitement est de pouvoir premièrement par la suite interpréter la structure de la phrase d'une manière grammaticale. Ensuite, donner un sens clair à certains mots qui parfois selon la structure grammaticale de la phrase où ils se trouvent peuvent avoir un tout autre sens.

Par exemple : « Je me rends au pied de la montagne », les labels de POS tagging correspondants seraient « (Je, Pronom), (me, Pronom), (rend, Verbe), (au, Préposition), (pied, Nom), (de, Préposition), (la, Déterminant), (Montagne, Nom) ».

## 3. Analyse Sémantique

### a. *Analyse de sentiments*

L'analyse de sentiments est une sorte de boite à outils dans laquelle il existe une multitude d'outils pour y parvenir mais que je ne vais pas élaborer sur. Le principe général de l'analyse de sentiments repose sur des dictionnaires de mots étant associés à des sentiments et ensuite passés au moteur NLP et si leur présence est détectée, selon la quantité présente de ces derniers ou même le manque de mots évoquant un sentiment, un ou plusieurs sentiments seront assignés au texte analysé. Il est possible dans certains cas d'associer des sentiments à des principes ou objets afin d'affiner l'analyse.

### b. *Named Entity Recognition*

Le NER ou Reconnaissance d'entités nommées est un des nombreux outils NLP permettant de donner davantage de sens aux tokens présents. Selon un dictionnaire

associant des mots à une catégorie prédefinie ou même une association de classes grammaticale commune avec une catégorie d'entité, l'ensemble de tokens sélectionnés se verra attitrer un type d'entité ou dans le cas contraire, aucun.

Par exemple : « La semaine dernière, le cours de la bourse chez Twitter a chuté de 4.8% ». Les entités ressorties de cette phrase sont : « (Semaine dernière, Date), (Twitter, Organisation), (4.8%, Pourcent) »

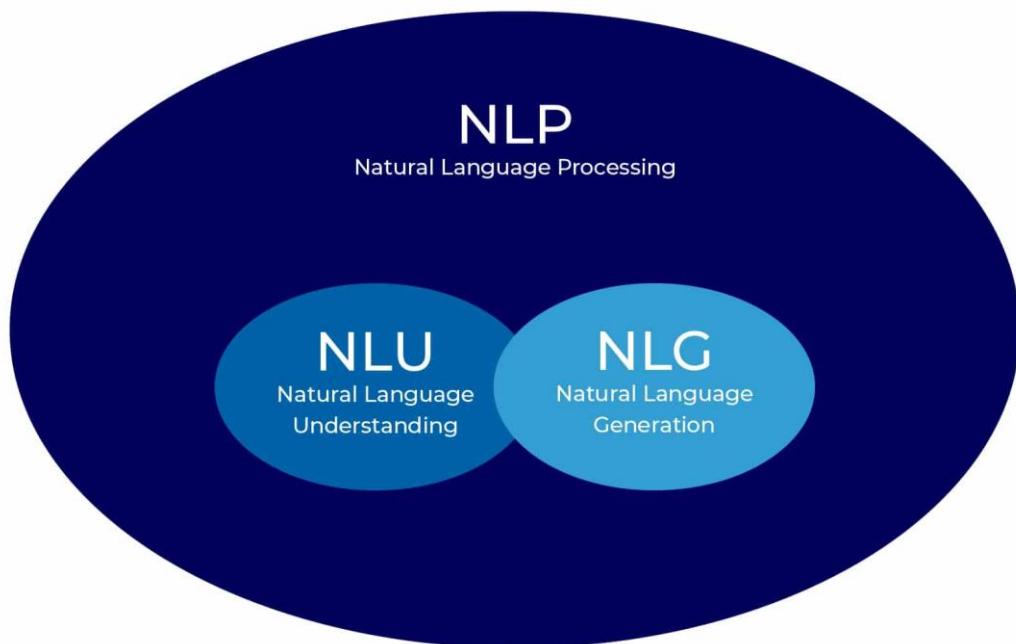
#### **4. Analyse Pragmatique**

L'analyse Pragmatique n'indique pas spécialement une technique en particulier mais consiste davantage en la combinaison des techniques précédemment utilisées afin de déterminer le contexte global du texte. L'ensemble des tokens et labels ressortis précédemment vont, une fois comparés entre eux à l'aide de dictionnaires et autre datasets permettant d'aider à la définition du lien entre eux, de donner un contexte.

#### **5. Intégration des données**

Cette dernière étape, l'intégration des données, est vitale dans le cadre d'autres techniques NLP comme la récupération d'information présente dans le texte, génération de résumé de textes ou l'extraction d'information présente de manière non structurée dans le texte.

Le NLP en tant que tel sert principalement à traiter du texte de manière générale pour être utilisé par la suite par d'autre algorithmes. Le NLP peut être vu comme la catégorie principale en matière de traitement de texte, cependant il existe deux sous catégories au NLP étant le Natural language Understanding ou Compréhension de langage Naturel (NLU) et le Natural Language Generation ou Génération de langage Naturel (NLG).



*Illustration 9: Représentation de la relation NLP, NLU et NLG*

Source : Datasolut, ref. : URL03

a) **NATURAL LANGUAGE UNDERSTANDING**

Selon Datasolut, “Le domaine du NLU s’occupe de la compréhension du langage naturel. Diverses méthodes de compréhension de texte sont utilisées à cet effet. Spécifiquement, la grammaire et le contexte de paires de mots ou mots uniques sont analysés afin d’en ressortir la signification de ces derniers et de la phrase. En outre, la sémantique, syntaxe, intention et émotion dégagée par un text sont examinés.”<sup>1</sup>

---

<sup>1</sup> WUTTKE, LAURENZ, 2023. NLP vs. NLU vs. NLG: UNTERSCHIEDE, FUNKTIONEN UND BEISPIELE. DATASOLUT GMBH [EN LIGNE]. 24 MAI 2023. DISPONIBLE A L’ADRESSE : [HTTPS://DATASOLUT.COM/NATURAL-LANGUAGE-PROCESSING-VS-NLU-VS-NLG-UNTERSCHIEDE-FUNKTIONEN-UND-BEISPIELE/](https://datasolut.com/natural-language-processing-vs-nlu-vs-nlg-unterschiede-funktionen-und-beispiele/) [CONSULTE LE 1 MARS 2024].

Deux concepts principaux pour parvenir à la compréhension de textes par NLU sont :

1. Détection de l'intention : Le processus d'identification des sentiments d'un utilisateur et de déterminer son objectif. Un des buts premiers du NLU étant l'établissement du sens du texte donné.
2. Détection d'entités : Un processus plus spécifique qui va chercher à identifier des entités spécifiques et d'extraire les informations les plus importantes de ces entités. Il existe deux types d'entités étant les entités nommées et numériques.

Les entités nommées sont groupées en catégories telles que des noms d'individus, de business et de lieux.

Les entités numériques sont reconnues en tant que quantités, dates, monnaies et pourcentages.<sup>2</sup>

Le NLU est une technologie prévalente dans le développement de Chatbots et autres assistants intelligents ayant pour but de converser avec un utilisateur et de comprendre ce que ce dernier lui réponds.

**b) NATURAL LANGUAGE GENERATION**

### **3.2. TAPAS**

---

<sup>2</sup> WHAT IS NATURAL LANGUAGE UNDERSTANDING (NLU)? | DEFINITION FROM TECHTARGET, ENTERPRISE AI [EN LIGNE]. DISPONIBLE A L'ADRESSE : [HTTPS://WWW.TECHTARGET.COM/SEARCHENTERPRISEAI/DEFINITION/NATURAL-LANGUAGE-UNDERSTANDING-NLU](https://www.techtarget.com/searchenterpriseai/definition/natural-language-understanding-nlu) [CONSULTE LE 1 MARS 2024].

a) **BERT**

b) **TAPAS : FONCTIONNEMENT**

### **3.3. RASA**

### **3.4. RASA PRO**

## **CONCLUSION (STYLE « TITRE 1 »)**

Votre texte, votre texte (style « Corps de texte, interligne 1,5 »).

## **ANNEXES (STYLE « TITRE 1 »)**

Imprimer idéalement cette page sur une page de couleur

Chaque annexe doit commencer sur une nouvelle page et doit être numérotée :

Annexe 1 puis Annexe 2, etc.

## **ANNEXE 1**

## **ANNEXE 2**

## **ANNEXE 3**

## **RÉFÉRENCES DOCUMENTAIRES (STYLE « TITRE 1 »)**