



UNIVERSITÉ  
CAEN  
NORMANDIE

MASTER INFORMATIQUE  
Internet, Données et Connaissances

Mémoire de stage

# Systèmes NLP et NLU en entreprise

Le traitement du langage naturel au service de l'utilisateur



Clément Vétillard

Tuteur de stage : **Marc Spaniol**  
Jury : **François Rioult**

Entreprise d'accueil : KMB Labs  
Maitre de stage : **Paul Leménager**  
Année universitaire : 2020 / 2021





# Table des matières

<b>Introduction</b>	<b>5</b>
<b>1 Contexte</b>	<b>6</b>
1.1 Les intentions . . . . .	6
<b>2 KMB Labs</b>	<b>8</b>
2.1 Les interfaces . . . . .	8
2.1.1 Le chatbot . . . . .	8
2.1.2 L'interface conversationnelle . . . . .	8
2.2 Le backoffice . . . . .	9
2.3 L'API . . . . .	9
2.4 Le Framework . . . . .	10
<b>3 Projets</b>	<b>11</b>
3.1 Les templates . . . . .	11
3.1.1 Objectif . . . . .	12
3.2 ML Monitor . . . . .	13
3.4 Interface conversationnelle . . . . .	13
<b>4 Conclusion</b>	<b>14</b>

# Introduction

KMB Labs<sup>1</sup> est une entreprise de développement web spécialisé dans les assistants conversationnels et moteurs de recherches. L'équipe propose à ses clients d'intégrer, au choix, un moteur de recherche spécifique au site client ainsi qu'un assistant chatbot.

Avec en tout 6 employés, KMB Labs possède à son actif plusieurs projets menés à bien dont la plupart sont connus du grand public, tant dans l'immobilier (*Guy Hoquet, Century 21*) que dans le recrutement (*Adecco, Carrefour, La Poste*).

Le domaine des entreprises visées par l'expertise de KMB Labs tant à se diversifier afin d'accueillir des chatbots spécialisés dans d'autres domaines (comme des chatbot interne à une entreprise mise à disposition pour ses employés).

*« Nous mettons en relation la dernière génération d'algorithmes de matching avec notre technologie d'analyse texte - et tout cela en une fraction de seconde. »*

**Vu sur [kmblabs.com](http://kmblabs.com)**

Lors de ce stage, plusieurs missions m'ont été confiées afin de participer à la mise en place de différentes fonctionnalités afin d'améliorer les applications déjà en place. Après une présentation du contexte et des différentes parties composant les services fournies par KMB Labs, je m'efforcerai de détailler mon implication dans chacun des projets auxquels j'ai pu participer pendant toute la durée de mon stage.

---

1. Anciennement Kick My Bot

# 1 Contexte

Notre époque connaît un nombre croissant d'entreprises choisissant de réaliser une vitrine sur le web et d'y proposer des services via celui-ci. Certains site peuvent devenir une structure conséquente de données, aussi bien dû aux nombreux services proposés qu'aux informations dont ils disposent. Même si certaines pratiques, comme les foires aux questions, tendent à regrouper les questions les plus fréquentes des utilisateurs, cela ne saurait regrouper efficacement toutes les informations d'un site.

Pour remédier à cette problématiques, plusieurs approches ont été utilisées. Les barres de recherches, indexant le contenu des différentes pages ainsi que des fenêtres permettant de communiquer avec un support sous la forme d'une discussion.

Cependant, la première approche n'est pas forcément intuitive pour les plus néophytes (là où on recherche une certaine information, il faudra choisir les bons mot-clefs pour obtenir le résultat escompté). La deuxième solution nécessite du personnel pour répondre aux questions des utilisateurs.

Avec l'émergence des systèmes de NLP<sup>1</sup> chacune de ses deux approches ont pu grandement évoluer. Ces outils permettent d'avoir une compréhension plus profonde de ce que recherche l'utilisateur afin de l'aiguiller vers le contenu qu'il recherche avec une plus grande facilité.

L'objectif ici est de permettre à l'utilisateur de faire des requêtes à une interface en formant des phrases ou bien en parlant via un micro. Cette solution permet d'avoir une solution conviviale pour répondre aux questions de l'utilisateur et de pouvoir le rediriger vers le contenu souhaité rapidement, sans mobiliser de personnel.

## 1.1 Les intentions

Les différentes parties de ce rapport seront étroitement liées aux intentions. Les différents services proposés par KMB Labs cherche à détecter l'intention de l'utilisateur pour lui fournir la réponse la plus appropriée.

---

1. Natural Language Processing

Ainsi, les différentes entités grammaticales<sup>2</sup> nécessaires à la réalisation de l'intention sont extraites. De fait, un utilisateur cherchant à contacter une personne de l'entreprise pourra formuler sa demande de plusieurs façons :

- « je peux joindre quelqu'un de chez vous ? »
- « vous avez un numéro de téléphone ? »
- « j'ai besoin de vous contacter, je fais comment ? »
- etc...

Toutes ses manières, et bien d'autres encore, permettent de rediriger vers une page précise contenant les informations, ou bien de donner les informations nécessaires à la prise de contact. L'important est que l'*intention* de l'utilisateur pour toutes ces phrases est la même.

---

2. Sujet, action, etc...

## 2 KMB Labs

Afin de procurer ses services, KMB Labs a développé plusieurs applications et services web. Faisons une courte présentation de chacun de ses projets afin d'appréhender l'organisation de ceux-ci.

### 2.1 Les interfaces

#### 2.1.1 Le chatbot

Le chatbot se présente comme une fenêtre de discussion, on peut y converser avec un assistant virtuel qui répondre à nos questions. En parlant naturellement via l'interface, l'intention de l'utilisateur y est détectée et une réponse lui ai donné.

Ladite réponse est préalablement fournie par le client afin de choisir quelles interrogations sont prises en charge, ou non, par l'assistant virtuel.

La fenêtre peut-être agrémentée d'un onglet avec des questions pré-établies, un peu à la manière des foires aux questions, d'actions rapide où l'utilisateur n'a qu'à cliquer dessus pour obtenir le résultat rapidement.

#### 2.1.2 L'interface conversationnelle

L'interface conversation prend la forme d'une barre de recherche sur une page complète. L'utilisateur peut décrire ce qu'il recherche (caractéristiques d'un appartement, les domaines de compétences d'un emploi, etc...) et le moteur se charge de :

1. voir si une intention est détectée (à la manière du chatbot)
2. effectuer la recherche en appliquant les filtres trouvés la requête (localisation, salaire, loyer, etc...)

Cette interface se présente plus comme un moteur de recherche, mais l'intégration des systèmes NLP permettent une certaines flexibilités quant à la forme de la demande de l'utilisateur. Elle simplifie également l'utilisation, là où différents filtres (toujours



existants) devaient être rempli manuellement, l'interface ici va remplir les filtres correspondant aux critères de recherche.

## 2.2 Le backoffice

Le backoffice est une application web permettant au client de paramétrer les réponses qui seront fournies à ses utilisateurs et de s'informer du bon fonctionnement du chatbot.

Dans cette optique, plusieurs pages sont disponibles. En voici une présentation non exhaustive :

- l'éditeur : permet de définir les réponses du chatbot pour chaque intention disponible dans le projet
- les statistiques : regroupent des graphiques visant à informer le client du taux d'utilisation et de compréhension de sa solution conversationnelle fournie par KMB Labs
- ML<sup>1</sup> Monitor : cette partie, pour utilisateur averti uniquement, permet de tester la compréhension d'une phrase et, si besoin, d'entraîner un modèle avec des phrases non comprises qui aurait dû l'être
- le Lab fourni une interface simple permettant de voir la configuration de la fenêtre du chatbot telle qu'elle serait dans le site de destination mis à part l'environnement graphique

## 2.3 L'API

L'API a été écrite avec le langage GraphQL avec le service AppSync de Amazon Web Service<sup>2</sup>. Cette approche permet de ne récupérer que les données nécessaires à l'usage, et ceux même si la structure est complexe.

AppSync donne la possibilité de lier le schéma à des *resolvers*, qui sont en charge de retourner la donnée voulu. Ce procédé permet une grande flexibilité, les résolveurs peuvent être associé à :

- des requêtes directes aux bases de données (via le langage VTL)
- des *Lambda* qui sont des fonctions dans le langage désiré (Node.js, Python, Go, Java, etc...<sup>3</sup>)

---

1. Machine Learning

2. Abrégé « AWS » ci-après

3. Source : <https://aws.amazon.com/fr/lambda/>

La possibilité de pouvoir lier un résolveur à une fonction permet d'y effectuer plusieurs traitements avec que les données ne soient retournées à l'application cliente. Conclusion, cela allège la complexité des applications utilisant les lambdas dans leur cycle de vie, ou plus précisément cela reporte une partie de la charge sur les serveurs d'AWS.

## 2.4 Le Framework

## 3 Projets

Lors de ce stage, plusieurs projets m'ont été confiés. Dans cette partie je détaillerai mes différentes interventions. J'y détaillerai l'objectif du projet ainsi que les différentes tâches qui m'ont été confiées.

### 3.1 Les templates

Le premier projet auquel j'ai participé, afin de me faire à l'agencement du Backoffice et de comprendre le fonctionnement des différents éléments qui le composent, avait pour objectif de paramétrer plus rapidement les projets.

Les projets, associé aux clients, doivent être lié à des modèles d'apprentissage pour permettre la compréhension d'un domaine. Faisons la comparaison entre l'immobilier et les ressources humaines.

Dans l'immobilier, le chatbot doit pouvoir comprendre les intentions liées loyer, prix d'achat et de vente, métrage, etc... De l'autre côté, un chatbot lié aux ressources humaines devra comprendre les intentions liés aux horaires de travail, au salaire, etc... En plus des intentions spécifiques au domaine du chatbot vient s'ajouter des intentions plus générales comme les formules de politesse pour saluer l'utilisateur, répondre à ses remerciements et plus encore.

Pour associer un modèle de compréhension à un chatbot, une personne de chez KMB Labs devaient jusqu'alors sélectionner manuellement dans les paramètres du projet ledit modèle.

N.B : ce projet est très diversifié dans sa manière de s'intégrer à l'existant, nous verrons que celui-ci touche à beaucoup d'aspect différents du backoffice.

### 3.1.1 Objectif

L'objectif d'ajouter des templates était de pouvoir facilement associer plusieurs modèle à un projet en sélectionnant un template. Pour se faire, deux points sont à prendre en compte :

- la création du template
- l'association à un projet

#### La création de template

Pour créer différents éléments du backoffice (utilisateur, projets, intention) un admin ouvre ce que l'on appelle une *popin*<sup>1</sup> en utilisant le bouton correspondant dans les paramètres.

Pour les templates, le même principe a été utilisé. Pour créer un template il faut donc ouvrir la *popin* ce qui ouvre l'interface suivante :

The screenshot shows a modal window titled "Gestionnaire de templates" with a close button (X) in the top right corner. The form is divided into sections for creating a new template. The first section, "Créer un nouveau template", contains a text input field for "Nom du template" and a dropdown menu labeled "Select..." with a green checkmark icon. Below this is a green button labeled "Ajouter un bouton". The next section, "Immobilier", features a horizontal list of tags: "Achat", "Coronavirus", "Finance", "Immobilier", "Juridique", and "Location", each with a small 'x' icon for removal. Below these tags are two more tags: "Problème location" and "Smalltalk", also with removal icons. A red square icon is visible below the tags. The following section, "RH", contains a horizontal list of tags: "Administration / Intérim", "Congés & Absences", "Contact", "Coronavirus", "Emploi et carrières", "Entreprise", "Formation", "Politique RH", and "Process RH", each with a removal icon. Below these are three more tags: "Présentation entreprise", "Site web", and "Smalltalk", also with removal icons. At the bottom of the form are two green buttons: "Annuler" on the left and "Valider" on the right.

L'objectif ici était donc d'ajouter une nouvelle *popin* à l'interface en utilisant l'existant. React étant orienté composant, il est très facile de réutiliser l'existant : le squelette de

---

1. interface pouvant s'afficher par dessus le reste de l'application web

la fenêtre était déjà réalisé, il ne me restait plus qu'à comprendre comment une *popin* était affichée.

En effet la gestion de celle-ci passe par un *store* Redux<sup>2</sup>. Le gestionnaire des *popins* est toujours affiché, cependant une *popin* n'est affichée seulement si son identifiant est présent dans le store.

## 3.2 ML Monitor

## 3.3

## 3.4 Interface conversationnelle

---

2. « *A Predictable State Container for JS Apps* » Vu sur [redux.js.org](https://redux.js.org)

## 4 Conslusion