République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Saad DAHLAB - Blida 1

Faculté des sciences

Département d'Informatique

Mémoire présenté par :

Mlles. ADJANI Nassiba et BENBAIBECHE Romeissa

Pour l'obtention du diplôme de Master

Domaine : Mathématique et Informatique

Filière : Informatique

Spécialité : Traitement Automatique de la Langue

Sujet :

Interrogation du System d'Information Décisionnel de l'ANEM en Langage Naturel

Soutenu le Jeudi 21 Janvier 2021, devant le jury composé de

Mr. Bougheda		Encadreur
Dr. A. MEZZI	Université de Blida 1	Promotrice
Pr. N. BENBLIDIA	Université de Blida 1	Présidente
Dr. ABBACHE	Université de Chlef	Examinateur

Résumé

Dans les entreprises, le système d'information décisionnel SID est dédié à l'interrogation et à l'analyse des données de l'entreprise principalement dans le but de prise de décision.

L'ANEM qui est l'agence nationale de l'emploi a mis en place un système d'information décisionnel qui permet d'avoir une nette visibilité sur les fluctuations du marché de l'emploi, notamment la contribution dans la prise de décision au niveau du gouvernement.

Cependant, il existe quelques contraintes liées à son utilisation par les end-user notamment une difficulté pour générer des commandes (requêtes) affinés via MDX ou outils Ad-hoc afin de répondre au besoin décisionnel ou analytique.

Pour répondre à la problématique posée, nous avons mis en place un système Q&A avec une interface web où l'end-user interroge le SID via un langage naturel plus aisé que l'interrogation via MDX. Ce système va faire la traduction de la requête en langage naturel vers une requête mdx la plus proche possible de l'attente de l'utilisateur.

Mots clés :

Question/ Réponse, Système d'information décisionnelle, Requête mdx.

Abstract

In companies, the SID decisional information system is dedicated to querying and analyzing company data mainly for decision-making purposes.

The ANEM, which is the national employment agency, has set up a decisional information system that provides clear visibility on fluctuations in the job market, including the contribution to decision-making at the government level.

However, there are some constraints related to its use by end-users, including a difficulty in generating refined commands (queries) via MDX or Ad-hoc tools in order to meet decision-making or analytical needs.

To address this issue, we have implemented a Q&A system with a web interface where the end-user queries the SID via a natural language that is easier than querying it via MDX. This system will translate the natural language query into a mdx query as close as possible to the user's expectations.

Keywords:

Q&A, decisional information system, mdx query.

ملخص

فـي الشـركات ، يـتم تخصـيص نظـام معلومـات اتخـاذ القـرار SID للاسـتعلام عـن بيانـات الشـركة وتحليلها بشكل أساسي لغرض صنع القرارات.

أنشـأت ANEM ، وهـي الوكالـة الوطنيـة لتشـغيل ، نظـام معلومـات لاتخـاذ القـرار يـوفر رؤيـة واضـحة للتقلبات في سوق الشغل ، و يساهم في صنع القرار على المستوى الحكومي.

ومـع ذلـك ، هنـاك بعـض القيـود المتعلقـة باسـتخدامه مـن قبـل المسـتخدمين النهـائيين ، بمـا فـي ذلـك صـعوبة إنشـاء أوامـر محسّـنة (اسـتعلامات) عبـرMDX او أدوات Ad-hoc مـن أجـل تلبيـة احتياجـات صنع القرار أو التحليل.

لمعالجـة هـذه المشـكلة ، قمنـا بتطبيـق نظـام سـؤال وجـواب بواجهـة ويـب حيـث يسـتعلم المسـتخدم النهـائي عن SID عبر لغة طبيعية أسهل من الاستعلام عنها عبر .MDX سيقوم هذا النظام بترجمة استعلام اللغة الطبيعية إلى استعلام MDX الاقرب الى تطلعات المستخدم.

<u>الكلمات المفتاحية :</u>

استعلام *MDX* ، نظام سؤال وجواب ،,نظام معلومات لاتخاذ القرار

Dédicaces

Avec l'expression de ma reconnaissance, je dédie ce modeste travail

A ceux qui, quels que soient les termes embrassés, je n'arriverais jamais à leur exprimer mon amour sincère.

A l'homme, mon précieux offre du dieu, qui doit ma vie, ma réussite et tout mon respect : mon cher père Smain.

A la femme qui a souffert sans me laisser souffrir, qui n'a épargné aucun effort pour me rendre heureuse : mon adorable mère Bachira.

A mon adorable sœur Zineb pour son soutien et à mon frère AbdelKarim et à ma petite sœur Chaima que Dieu les protège et leurs offre la chance et le bonheur.

Merci pour leurs amours et leurs encouragements.

Sans oublier mon binôme Romaissa pour son soutien moral, sa patience et sa compréhension tout au long de ce projet. Ainsi qu'a toutes mes amies et spécialement Yasmine Ikram et Mehdia.

ADJANI Nassiba

Dédicaces

Je dédie ce modeste travail avec un grand plaisir aux êtres les plus chers à moi :

A mon cher PAPA, qu'il a toujours été à mes côtés pour me soutenir et m'encourager, qui n'a jamais dit non à mes exigences. Que ce travail traduit ma gratitude et mon affection.

A ma chère MAMAN, une source d'amour, qui a souffert sans me laisser souffrir, nulle dédicace ne peut exprimer ce que je lui dois.

A mes très chers frères pour leurs soutiens moraux et encouragements

A mes adorables amies Rania, Yasmine, Hadjer, Sarah, et Nayel pour leurs soutiens et leurs conseils tout au long de mes études.

A mon très chère amie et camarade de travail Nassiba pour ses efforts considérables, sa patience et sa compréhension tout au long de ce projet.

A toute ma famille et ceux qui m'ont aidé à réaliser ce travail.

A tous ceux que j'aime et qui m'aiment.

BENBAIBECHE Romaissa

Remerciement

Tout d'abord, Nous tenons à remercier ALLAH le tout puissant et le miséricordieux, qui nous a donné la force et la patience d'accomplir ce modeste travail.

Nous tenons à adresser également nos remerciements à notre encadreur M. AHMED BOUGHEDDA et Mlle AHLEM BOUKAIOU qui ont bien voulu mettre leurs incomparable savoir et expériences à notre disposition.

Nos remerciements vont ainsi tout spécialement à nos familles, qui ont sus nous supporter et encourager tout au long de notre vie, ainsi que pour leur aide inestimable, leur patience et leur soutien indéfectible.

Nous tenons aussi, à remercier tous les enseignants qui ont contribué de près ou de loin à notre formation spécialement Mme.MELYARA MEZZI pour sa bonne volonté d'accepter de nous encadrer, pour tout le temps qu'elle nous a octroyé et pour tous les conseils qu'elle nous a prodiguée.

On remercie vivement Messieurs les membres du jury d'avoir accepté d'évaluer ce travail.

Pour finir, et afin de n'oublier personne (amis, membre de la famille et tous ceux qui nous sont chers)

Nassiba et Romaissa

Table des matières

Introduction générale 1

1. Contexte global 2

2. Présentation de l'organisme d'accueil [1].....2

2.1. Présentation de l'ANEM 2

2.2. Missions de l'ANEM.....3

2.3. Organisation de l'ANEM 4

2.4. Organigramme de l'ANEM.....4

3.Problématique 6

4.Objectifs 6

5. Organisation du mémoire.....6

Chapitre I : Les systèmes Questions/ Réponses.....7

1. Introduction.....8

2. L'intelligence Artificielle.....8

3. Le traitement automatique des langues :.....8

4. Systèmes Question/ Réponse 9

4.1. La structure d'un système Q&A.....9

4.1.1. Module d'analyse des questions 9

4.1.2. Module de recherche de paragraphes..... 10

4.1.3. Module d'extraction de réponses..... 10

4.2. La classification des systèmes de réponse aux questions..... 11

4.2.1. Classification basée sur le domaine 11

4.2.2. Classification basée sur les types de questions 12

4.2.3. Classification basée sur la source de données..... 13

4.2.4. Classification basée sur les formes de réponses générées par le système Q&A : 15
4.2.5. Classification basée sur le paradigme du langage : 16
4.2.6. Classification basée sur des approches 17
5. Chatbot : 19
5.1. La classification des Chatbots 19
5.1.1. Classification basée sur le domaine de connaissances 20
5.1.2. Classification basée sur le service fourni 20
5.1.3. Classification basée sur les objectifs 21
5.1.4. Classification basée sur les entrées/ sorties : 22
6. Conclusion 24
Chapitre II : Entrepôts de données 25
1. Introduction 26
2. Système d'information décisionnel 26
3. Informatique décisionnelle 26
4. Chaîne décisionnelle [10] 27
4.1. Collecte et alimentation 28
4.2. L'intégration 28
4.3. Organisation 29
4.4. Restitution 29
4.4.1. Les outils de reporting et de requêtes 29
4.4.2. Les outils d'analyse OLAP 30
4.4.3. Les outils de Datamining 30
5. Entrepôt de données (DW) 31
5.1. Caractéristiques d'un entrepôt de données 31

6. La modélisation multidimensionnelle 32
6.1. Modèles de représentation des données 32
6.1.1. Concepts fondamentaux 32
6.1.2. Modélisation conceptuelle 33
6.1.3. Modélisation Logique 34
7. OLAP 36
7.1. Structure OLAP 36
7.2. Architecture d'OLAP 37
7.3. Opérations de manipulation OLAP : 38
7.4. Comparaison entre OLAP et OLTP 40
7.4.1. OLTP 40
7.4.2. Comparaison 40
7.5. MDX pour l'interrogation d'OLAP 41
7.5.1. Définition : 41
7.5.2. Structure et utilisation de la requête MDX 41
7.5.3. Comparaison entre SQL et MDX : 42
8. Conclusion 44
Chapitre III : Conception et modélisation de la solution 46
1. Introduction 47
2. Solution proposée 47
2.1. Architecture fonctionnelle 47
2.2. Architecture technique 48
3. Modélisation 49
3.1. Diagramme d'activités 49
4. Conclusion 53

Chapitre IV : Implémentation de la solution.....54
1. Introduction.....55
2. Environnement de développement.....55
2.1. Outils.....55
- Django.....55
2.2. Langages de programmations.....56
3.Mise en œuvre.....57
3.1. Création du chatbot.....57
3.1.1. Intentions (Intent).....57
3.1.2. Entités.....58
3.1.3. Phrases d'entraînement (Training phrases).....60
3.2. Constitution de la requête MDX.....61
3.2.1. Récupération des paramètres.....61
3.2.2. Classification des mesures et dimensions.....62
3.2.3. Partie 1 : Sélection sur les mesures.....62
3.2.4. Partie 2 : Projection sur les dimensions.....63
3.2.5. Finalisation de la requête.....63
3.3. Exécution de la requête.....64
3.4. Interface web.....64
4.Conclusion.....67
Conclusion générale.....68
1. Conclusion générale :.....69
2. Perspectives :.....70
Références bibliographiques.....71

Liste des figures :

Figure 1: Organigramme de l'ANEM. 5

Figure 2: Structure d'un système Q&A. 9

Figure 3: Classification des systèmes Q&A. 11

Figure 4: Classification des Chatbots. 20

Figure 5: modélisation de la chaine décisionnelle. 27

Figure 6: Les étapes du processus ETL. 28

Figure 7: Modèle multidimensionnel 32

Figure 8: Modélisation en étoile 33

Figure 9:Modélisation en flacon 34

Figure 10: Modèle en constellation 34

Figure 11: Architecture ROLAP 35

Figure 12: Architecture MOLAP 35

Figure 13: Architecture HOLAP 36

Figure 14 : Architecture typique d'un système OLAP. 38

Figure 15: Opération de manipulation : Drill up/Drill Down 39

Figure 16: Opération de manipulation : Rotation 39

Figure 17 : Opération de manipulation : Sélection 39

Figure 18: Opération de manipulation Projection 40

Figure 19 : Représentation du traitement d'un exemple. 48

Figure 20: Interaction entre les parties du système. 48

Figure 21 : Diagramme de cas d'utilisation .... **Erreur ! Signet non défini.**

Figure 22: Diagramme d'activités du système. 52

Figure 23: Diagramme de séquences 53

Figure 24: Analyse de l'orientation. 58

Figure 25: Groupe d'entités 59

Figure 26: Entité "Genre".....60

Figure 27 : Exemple de phrase d'entrainement avec paramètres 61

Figure 28: code de classification des mesures et dimensions 62

Figure 29: Fragment de code de la sélection sur les mesures.....63

Figure 30:lignes de code de la projection sur les dimensions.....63

Figure 31: lignes de code de la finalisation de la requête 64

Figure 32 Page principale.....65

Figure 33 Page d'erreur.....66

Figure 34 Page de résultats 66

Liste des tableaux

Tableau 1: tableau récapitulatif des différences entre OLTP et OLAP......41

Tableau 2: Comparaison entre SQL et MDX......43

Liste d’acronymes

BI : Business Intelligence

DW : Data Warehouse

DM: Data Mart

ETL: Extract Transform Load

OLAP: On-Line Analytical Processing

OLTP: On-Line Transaction Processing

SGBDR : Système de Gestion de Base de Données Relationnelle

SQL: Structured Query Language

SSAS: SQL Server Analysis Service

HOLAP : Hybrid OLAP.

MDX : Multidimensionnel expression.

MOLAP : Multidimensionnel OLAP

ROLAP: Relationnel OLAP

EDW:Entrepôt de Données d'Entreprise

ODS: Magasin de Données Opérationnel

MD: Modèle Multidimensionnel

SID : Systême d'information décisionnel

BI : Business Intelligence

Introduction générale

1. Contexte global

Nous vivons dans un monde axé sur les données, où une énorme quantité de données est collectée et stockée quotidiennement. Dans son rapport sur la numérisation mondiale, IDC estime que la création mondiale de données passera à 175 zettaoctets d'ici 2025, soit dix fois la quantité de données produites en 2017.

Plus il y a de données générées, plus il devient important d'avoir la capacité d'y accéder et de les analyser afin de les utiliser efficacement. Le domaine de l'informatique décisionnelle (Appelé BI) a pour but d'apporter des méthodes et des outils pour assister les utilisateurs dans leur tâche de recherche d'information.

L'ANEM qui est l'agence nationale de l'emploi dispose d'un système d'information décisionnel qui permet d'avoir une nette visibilité sur les fluctuations du marché de l'emploi, notamment la contribution dans la prise de décision au niveau du gouvernement.

Pour combiner les informations provenant de sources divers et hétérogène l'ANEM stocke ses informations dans un type spécial de base de données qui est l'entrepôt de donnée (data warehouse). Ces informations sont organisées et analysées grâce au serveur OLAP.

2. Présentation de l'organisme d'accueil [1]

2.1. Présentation de l'ANEM

Dans tous les pays du monde la gestion et la régulation du marché du travail relèvent des prérogatives de l'état : par un service public.

En application des textes de l'Organisation Internationale du Travail (OIT), un service public de l'emploi est instauré en Algérie par le décret n° 62-99 du 29 novembre 1962. Il est confié à l'Office Nationale de Main

d'Œuvre (ONAMO). Le dispositif sera ensuite remanié ou complété par des textes réglementaires ou législatifs à plusieurs reprises, notamment en 1963 (instauration d'un monopole sur les flux migratoires), 1971 (organisation de l'ONAMO). Avec l'arrivé de l'ONAMO en tant qu'institution chargée de l'emploi au niveau national, nous commençons à prendre en charge les différents mécanismes de gestion de l'emploi, des statistiques ont commencé à voir le jour et les placements des travailleurs connaissent une certaine dynamique. Des BMO (Bureau de Main – d'Œuvre) étaient ouverts à travers les différentes wilayas et communes qui permettaient de couvrir les besoins en main d'œuvres des secteurs industriels et notamment des services.

En 1990, un changement de dénomination de l'ONAMO en Agence Nationale de l'EMploi, (ANEM) est décidé. Elle est dotée en 2004 du statut d'établissement public à gestion spécifique. Elle est placée sous la tutelle du ministère du travail et de la sécurité sociale.

2.2. Missions de l'ANEM

Dans tous les pays du monde la gestion et la régulation du marché du travail relèvent des prérogatives de l'état : par un service public.

En application des textes de l'Organisation Internationale du Travail (OIT), un service public de l'emploi est instauré en Algérie par le décret n° 62-99 du 29 novembre 1962. Il est confié à l'Office Nationale de Main d'Œuvre (ONAMO). Le dispositif sera ensuite remanié ou complété par des textes réglementaires ou législatifs à plusieurs reprises, notamment en 1963 (instauration d'un monopole sur les flux migratoires), 1971 (organisation de l'ONAMO). Avec l'arrivé de l'ONAMO en tant qu'institution chargée de l'emploi au niveau national, nous commençons à prendre en charge les différents mécanismes de gestion de l'emploi, des statistiques ont commencé à voir le jour et les placements des travailleurs connaissent une certaine dynamique. Des BMO (Bureau de Main – d'Œuvre) étaient ouverts à travers les différentes wilayas et communes qui permettaient de couvrir les besoins en main d'œuvres des secteurs industriels et notamment des services.

En 1990, un changement de dénomination de l'ONAMO en Agence Nationale de l'EMploi, (ANEM) est décidé. Elle est dotée en 2004 du statut d'établissement public à gestion spécifique. Elle est placée sous la tutelle du ministère du travail et de la sécurité sociale.

2.3. Organisation de l'ANEM

L'ANEM possède un réseau territorial comprenant :

- 218 Agences Locales de l'EMploi (ALEM) ;
- 48 Agences de Wilaya de l'EMploi (AWEM) ;
- 11 Directions Régionales de l'EMploi (DREM).

Les agences locales, complémentaires des agences de wilaya, sont constituées en fonction des besoins locaux, environ 151 à la fin de 2009 et elles sont à 218 à la fin de 2015.

La direction générale, située à Alger, est confiée depuis 2011 à Mohammed Tahar CHALAL.

2.4. Organigramme de l'ANEM

La figure ci-dessous, représente l'organigramme général de l'ANEM

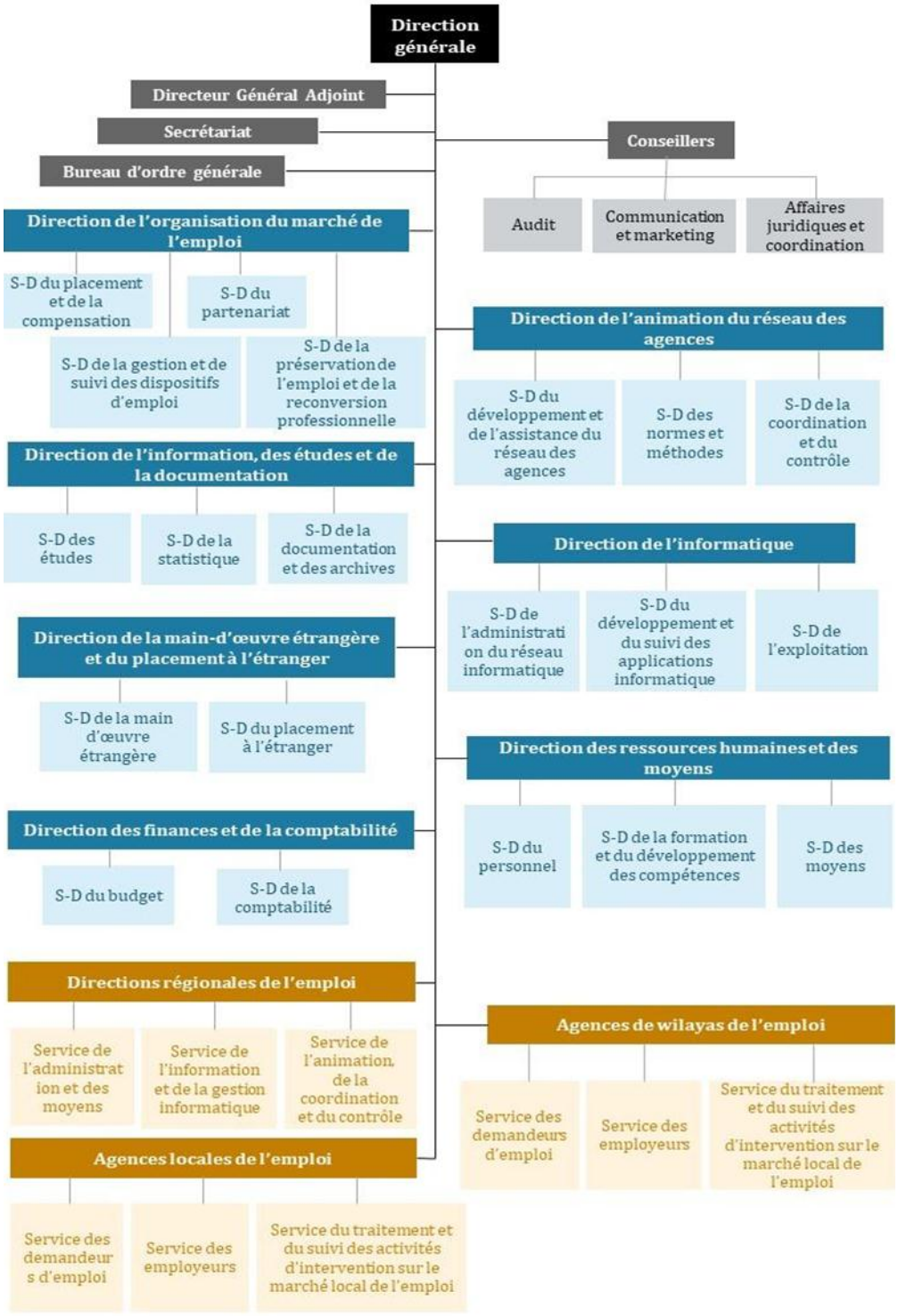


Figure 1: Organigramme de l'ANEM.

3.Problématique

Pour interroger les données, ces outils doivent pouvoir être utilisés par les non experts en bases de données. En effet, de tels outils ne sont pas naturels dans le sens où les utilisateurs doivent savoir comment les données sont organisées conceptuellement dans le modèle de données, afin de formuler des requêtes significatives.

Un sondage effectué en 2019 sur l'utilisation du SID a reflété quelques contraintes liées à son utilisation par les end-user. Parmi lesquelles la difficulté de générer des commandes (requêtes) affinés via MDX ou outils Ad-hoc afin de répondre au besoin décisionnel ou analytique.

4.Objectifs

L'utilisateur à tendance à interroger le SID en prenant en considérations plusieurs paramètres en relation avec son activité et environnement, cette situation laisse l'end-user interroger le SID via un langage naturel plus aisé et précis que l'interroger via MDX ou via une interface Ad-hoc.

Interroger le système OLAP du SID en langage naturel permet de générer une requête MDX la plus proche possible de l'attente ou besoin analytique attendu par l'utilisateur.

5. Organisation du mémoire

Notre mémoire est organisé en quatre chapitres : le premier concerne les systèmes questions réponses et la notion des chatbots et leur classification. Le deuxième chapitre synthétise les concepts liés aux systèmes d'information d'aide à la décision, ou nous avons évoqué la chaine décisionnelle, les systèmes multidimensionnelles, les outils OLAP et le langage MDX. Le troisième chapitre est consacré à la conception et la modélisation de notre solution proposée. Le chapitre 4 concerne la réalisation et l'implémentation de la solution proposé tout en spécifiant l'environnement de travail.

Chapitre I : Les systèmes Questions/ Réponses

1. Introduction

La quantité d'information ne cesse d'augmenter. Ainsi le besoin à un accès rapide et fiable à l'information a fait naître la nécessité de fournir aux utilisateurs finaux un accès à ces informations via des systèmes dont le système Question réponse fait part.

Dans ce chapitre nous identifions l'intelligence artificielle ainsi que le traitement automatique des langages, puis nous présentons le système questions réponses, son architecture ainsi que ses classifications, en suite nous aborderons la notion des chatbots, en donnant leur définition et classification.

2. L'intelligence Artificielle

L'intelligence artificielle (IA) fait référence à la simulation de l'intelligence humaine dans des machines programmées pour penser comme des humains et imiter leurs actions. Le terme peut également être appliqué à toute machine présentant des traits associés à un esprit humain tels que l'apprentissage et la résolution de problèmes.

La caractéristique idéale de l'intelligence artificielle est sa capacité à rationaliser et à prendre des mesures qui ont les meilleures chances d'atteindre un objectif spécifique [2].

3. Le traitement automatique des langues :

Le traitement automatique des langues naturelles (TALN) ou de la langue (TAL). Également connu sous le nom de NLP (Natural Language Processing en anglais) est un domaine de recherche qui se positionne à l'intersection de plusieurs disciplines : Intelligence artificielle, Informatique théorique, calcul statistique, linguistique, …etc. Dont le principal objectif est la conception et le développement de programmes capables de traiter de manière automatique des données linguistiques c'est-à-dire des données exprimées dans une langue dite naturelle [3]

4. Systèmes Question/ Réponse

La réponse aux questions (Q&A pour Question/ Answer en Anglais) est une discipline informatique dans les domaines de la recherche d'informations et du traitement du langage naturel (TAL), qui se concentre sur la construction de systèmes qui répondent automatiquement aux questions posées par les humains dans un langage naturel. Une compréhension informatique du langage naturel consiste en la capacité d'un système de programme à traduire des phrases en une représentation interne de sorte que ce système génère des réponses pertinentes aux questions posées par un utilisateur. [4]

4.1. La structure d'un système Q&A

Les systèmes de réponse aux questions suivent généralement une structure de pipeline avec trois modules majeurs à savoir : Analyse des questions, Récupération de passage et extraction de réponse. La figure suivante montre ce processus. [5]

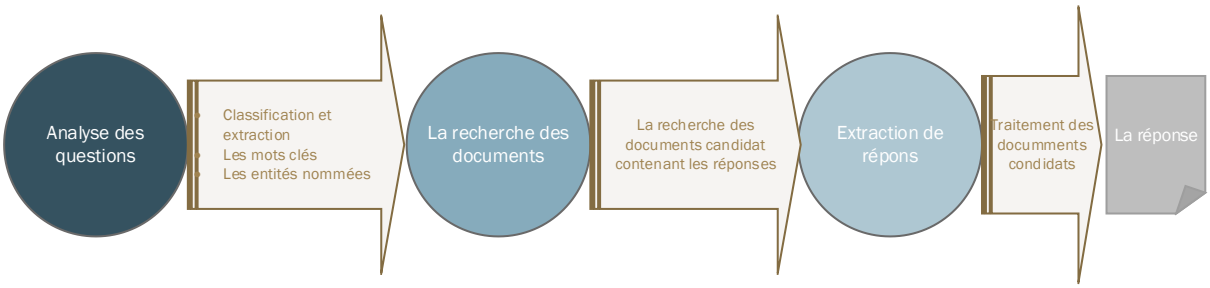


Figure 2: Structure d'un système Q&A.

4.1.1. Module d'analyse des questions

En règle générale, les questions posées aux systèmes Q&A doivent être analysées et comprises avant que des réponses puissent être trouvées. Par conséquent, tous les traitements de questions nécessaires sont effectués dans le module Analyse des questions. L'entrée pour cette étape est la requête de l'utilisateur en langage naturel et la sortie sont des représentations de la

requête. A ce niveau, les informations sémantiques contenues dans la requête, les contraintes et les mots-clés nécessaires sont extraits.

Les taches de ce module comprennent : l'analyse, la classification des questions et reformulation des requêtes. Ça signifie que la requête est analysée dans le but de représenter les principales informations requises afin de répondre à la requête posée par l'utilisateur ; la question est classée en fonction du mot-clé ou de la taxonomie utilisée dans la requête, ce qui conduit au type de réponse attendu ; la requête est également reformulée pour améliorer la formulation des questions et transformée en requêtes sémantiquement équivalentes, ce qui facilite le processus de recherche d'informations.

4.1.2. Module de recherche de paragraphes

La recherche de passages est naturellement basée sur un moteur de recherche orthodoxe pour récupérer un ensemble de passages ou de phrases candidats significatifs à partir d'une base de connaissances. Cette étape utilise les requêtes formulées à partir du module d'analyse des questions et recherche des sources d'informations pour trouver des réponses adaptées aux questions posées. Les réponses candidates provenant de sources dynamiques telles que le Web ou des bases de données en ligne peuvent également être incorporées ici.

Les structures de récupération de texte divisent le processus de récupération en trois étapes traitement, récupération, et classement.

L'étape de traitement implique l'utilisation d'analyseurs de requêtes pour identifier les textes dans une base de données. Ensuite, la récupération est effectuée en faisant correspondre des documents avec une ressemblance des modèles de requête. Le classement des textes renvoyés est ensuite effectué à l'aide de fonctions de classement comme tf-idf.

4.1.3. Module d'extraction de réponses

L'extraction de réponses fait partie intégrante d'un système Q&A Il produit la réponse exacte à partir des passages générés. Il fait cela d'abord par

produire un ensemble de réponses candidates à partir des passages générés, puis classer ces réponses à l'aide de certaines fonctions de notation.

Des études antérieures sur l'extraction de réponses ont discuté de l'utilisation de différentes techniques d'extraction de réponses, y compris les n-grammes, les modèles, les entités nommées et les structures syntaxiques.

4.2. La classification des systèmes de réponse aux questions

Divers travaux ont classé les systèmes de réponses aux questions, les visualisant sous différents points de vue et les classant selon différents critères. [5]

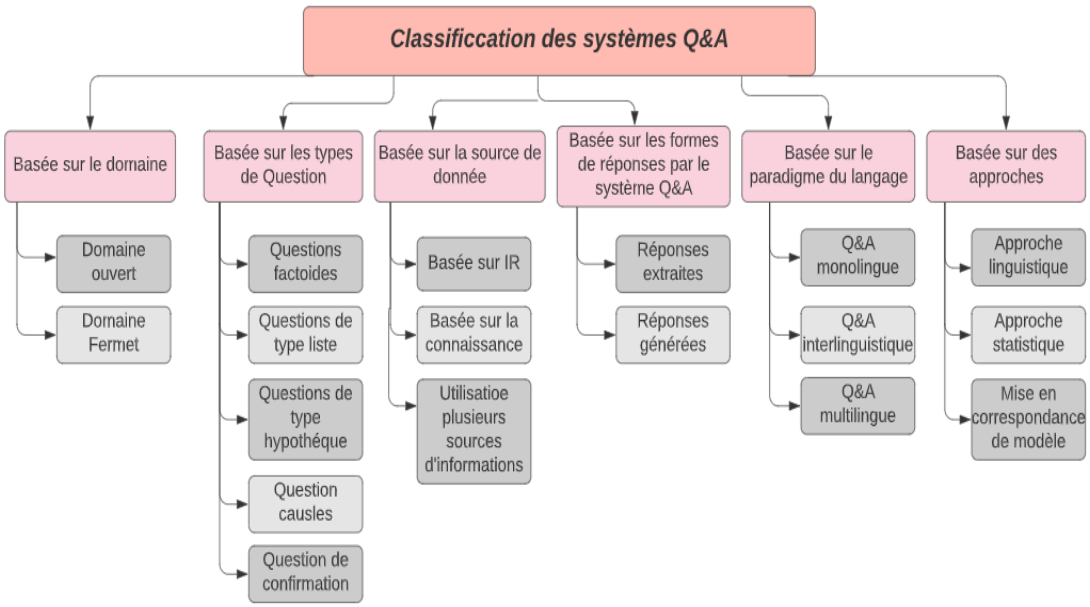


Figure 3: Classification des systèmes Q&A.

4.2.1. Classification basée sur le domaine

Il existe deux types de systèmes QA selon le domaine, domaine ouvert et domaine fermé ou restreint. Certaines autres revues les traitent respectivement en tant que système de réponse aux questions du domaine général et du domaine restreint.

- **Domaine ouvert :**

Fournit des réponses à toutes les questions, par exemple le système Google Knowledge n'a pas de restrictions de sujet, ainsi que de nombreux autres. Les questions générales du domaine ont un vaste référentiel de requêtes qui peuvent être posées.

- **Domaine fermé :**

  Offre des réponses sur certains sujets fixes, par exemple le système BASEBALL répond aux questions sur les matchs de baseball. Ce domaine nécessite une disposition linguistique pour comprendre le texte en langage naturel afin de fournir une solution précise aux requêtes.

  Les systèmes Q&A à domaine fermé peuvent être combinés pour créer un système Q&A à domaine ouvert.

4.2.2. Classification basée sur les types de questions

La génération de réponses aux requêtes des utilisateurs est directement liée au type de question posée. Par conséquent, la taxonomie des requêtes posées dans le système Q&A affecte directement les réponses. Sorties QA system révèlent que 36,4% des erreurs sont dues à une mauvaise classification des requêtes dans les systèmes Q&A. Des études catégorisent les requêtes dans une classification basée sur différents critères, parmi lesquels une classification selon tous les types de questions possible.

- **Questions factoïdes :** [quoi, quand, qui, qui, comment] :

  Ces questions sont de nature factuelle et renvoient à une seule réponse. Les requêtes factoïdes commencent généralement par « wh-word »

- **Questions de type liste** :

  La réponse à une requête de liste est une énumération. Les énumérations sont généralement une énumération d'entités ou de faits dans les réponses. La définition de la valeur de seuil pour la question de liste reste un problème dans ce système.

- **Questions de type définition :**

  Les types de questions de définition nécessitent un traitement complexe des documents récupérés, la réponse absolue consiste en un

passage de texte ou acquise après avoir résumé plus de documents. Habituellement, ce type de questions commence par « ce qui es ?» « what is » en anglais.

- **Questions de type hypothétique :**

  Cela nécessite des informations associées à tout événement supposé. Habituellement, cela commence par « Que se passerait-il si » les solutions sont subjectives à ces requêtes. Pas de bonne réponse spécifique pour les requêtes.

- **Questions causales [comment ou pourquoi] :**

  Les requêtes causales nécessitent des clarifications sur les entités qu'elles contiennent. Ces questions cherchent des explications, des raisons ou des précisions sur des événements ou des objets spécifiques.

- **Questions de confirmation :**

  On s'attend à ce que les solutions aux questions de confirmation se présentent sous la forme OUI ou NON.

  Un mécanisme d'inférence, un raisonnement de bon sens et une connaissance du monde sont nécessaires aux systèmes Q&A pour générer des réponses à la question causale

4.2.3. Classification basée sur la source de données

Sur la base de la source de données, la classification des systèmes QA est effectuée en trois catégories [6]:

- **Réponse aux questions basée sur IR**

  Ce system QA fournit une réponse à la requête de l'utilisateur en récupérant des extraits de texte du Web ou des corpus, les méthodes IR extraient des passages directement de ces documents, guidées par la question posée

  Le but est de répondre à la question d'un utilisateur en trouvant de courts segments de texte. Dans la phase de traitement des questions, un certain nombre d'informations de la question sont extraites. Le type de

réponse spécifie le type d'entité dont se compose la réponse (personne, lieu, heure, etc.). La requête spécifie les mots-clés à utiliser par le système IR pour rechercher des documents.

- **La réponse à une question basée sur la connaissance :**

  C'est de répondre à une question en langage naturel en la mappant à une requête sur une base de données structurée. La forme logique de la question se présente donc soit sous la forme d'une requête, soit facilement convertible en une seule. La base de données peut être une base de données relationnelle complète ou des bases de données structurées plus simples comme des ensembles de triplets RDF. Les systèmes de mappage d'une chaîne de texte vers n'importe quelle forme logique sont appelés analyseurs sémantiques. Les analyseurs sémantiques pour les réponses aux questions correspondent généralement à une version du calcul des prédicats ou à un langage de requête comme SQL ou SPARQL.

Pour effectuer le mappage à partir du texte (langage naturel question) sous une forme logique, la base de connaissance QA utilise certaines de ces méthodes :

- **La méthode basée sur des règles :** Cela se concentre sur le développement de la création manuelle règles pour extraire les associations fréquentes de la requête.
- **Méthodes supervisées :** Cela implique l'utilisation de données d'entraînement, qui contiennent paires de questions et leurs formes logiques, puis création d'un modèle qui mappe les questions à sa forme logique.
- **Méthode semi-supervisée :** Ensembles de données supervisés pouvant représenter pleinement toutes les formes de questions possibles dans lesquelles les requêtes peuvent être insérées n'ont pas encore été réalisées et c'est pour cette raison que la redondance textuelle est exploitée par la plupart des méthodes

pour mapper les requêtes sur des relations canoniques ou d'autres structures de la connaissance.

- **Utilisation de plusieurs sources d'informations :**

Cela implique l'utilisation de bases de connaissances structurées et d'ensembles de données textuelles pour fournir des réponses aux requêtes.

Le system DeepQA est basé sur le traitement de la question jusqu'à ce que la question soit classer par type exemple : question de définition, question de choix multiple, puzzle…

Ensuite DeepQA génère des réponses candidates en fonction du type de question, où la question traitée est combinée avec des documents externes et d'autres sources de connaissances pour suggérer de nombreuses réponses candidates qui peuvent être extraites de documents texte ou de bases de connaissances structurées. Ces réponses candidates passe par la suite sur des étapes de fusion et classement d'une manière itérative afin d'extraire la bonne réponse.

4.2.4. Classification basée sur les formes de réponses générées par le système Q&A :

Il existe deux catégories de réponses : extraites et générées.

- **Réponses extraites :**

Les réponses extraites sont divisées en trois catégories : Réponses sous forme de phrases, paragraphe et multimédia.

Les réponses sous forme de phrases dépendent des documents récupérés qui sont divisés en phrases individuelles, la phrase la mieux classée est renvoyée à l'utilisateur. Habituellement, des questions factoïdes ou de confirmation font partie de cette catégorie. Les réponses sous forme de paragraphes dépendent également des documents récupérés qui sont divisés en paragraphes individuels, le paragraphe qui se qualifie le plus comme réponse est présenté à

l'utilisateur. Les requêtes causales ou hypothétiques font partie de ce groupe. L'audio, la vidéo ou le clip sonore sont des fichiers multimédias qui peuvent être présentés comme réponses aux utilisateurs.

- **Réponses générées :**

  Les réponses générées sont classées en réponses conformationnelles (oui ou non), réponses d'opinion ou réponses de dialogue. Les questions de confirmation ont généré des réponses sous forme de oui ou de non, via la confirmation et le raisonnement. Les réponses d'opinion sont créées par système Q&A qui attribue des étoiles à l'objet ou aux caractéristiques de l'objet. Le système Dialogue Q&A renvoie les réponses aux questions posées sous forme de dialogue.

4.2.5. Classification basée sur le paradigme du langage :

La classification basée sur le paradigme du langage divise les systèmes QA par le nombre de langages utilisés dans le traitement des requêtes en trois catégories :

- **Systèmes de réponse aux questions monolingues :**
  Dans ce système, la requête de l'utilisateur, les documents de ressources et la réponse du système sont exprimés dans une seule langue.
- **Systèmes de réponse aux questions inter-linguistique :**
  Dans ce système, la requête de l'utilisateur et les documents de ressources sont exprimés dans des langues différentes et la requête est interprétée dans la langue des documents de ressources avant la recherche. Google Knowledge Graph traduit les requêtes saisies dans d'autres langues en anglais et renvoie une réponse également en anglais.
- **Systèmes de réponse aux questions multilingues :**
  Dans ce système, la requête de l'utilisateur et les documents de ressources sont exprimés dans la même langue et la recherche de requête est effectuée dans la même langue que celle dans laquelle la

requête a été exprimée. Cette classification est utile pour un utilisateur qui recherche des informations dans une langue spécifique qu'il ne connaît pas.

4.2.6. Classification basée sur des approches

Ci-dessous, les différentes approches de classification :

- **Approche linguistique :**

  Il est possible de réaliser une analyse syntaxique complète d'une collection de textes grâce à l'utilisation de techniques linguistiques de connaissances telles que le marquage de la parole, la tokenisation, l'analyse et la lemmatisation, et avec ces capacités, il devient réaliste d'exploiter les informations linguistiques pour une utilisation dans les systèmes de réponse aux questions, en particulier comme base de données pour IR.

- **Approche statistique :**

  La disponibilité d'une énorme quantité de données sur Internet a amplifié l'importance des méthodes statistiques, car cette approche met en avant des techniques qui peuvent traiter de très grandes quantités de données et les diversités que possèdent les données (hétérogénéité). Les méthodes statistiques sont libres de langages de requête structurés et peuvent créer des questions au format NL. Cette approche nécessite suffisamment de données pour un apprentissage statistique exact et elle produit de meilleurs résultats que les autres méthodes rivales une fois correctement apprises.

  Les méthodes statistiques traitent toujours chaque terme d'une requête individuellement et ne parviennent pas à reconnaître les caractéristiques linguistiques pour joindre des mots ou des phrases, ce qui est l'un de leurs principaux inconvénients.

- **Approche de mise en correspondance de modèle :**

La méthode d'appariement de motifs traite de l'influence communicative du texte modèle. Il remplace le traitement élégant utilisé dans d'autres méthodes informatiques.

De nombreux systèmes QA apprennent automatiquement les structures de texte à partir de passages au lieu d'utiliser des connaissances linguistiques complexes pour récupérer des réponses. La simplicité de tels systèmes le rend tout à fait favorable pour une mise en œuvre et une utilisation de petite taille.

De plus en plus les systèmes de correspondance de modèles de réponses aux questions utilisent un modèle de texte de surface, tandis que d'autres reposent sur des modèles pour le générateur de réponses. L'approche basée sur le modèle de surface consiste à trouver des réponses à des questions factuelles tandis que les modèles sont utilisés pour les systèmes à domaine fermé.

- **Basé sur le modèle de surface :**

  Cette méthode fait correspondre les réponses de la structure superficielle des textes récupérés en utilisant une longue liste de modèles générés (cela peut être fabriqué à la main ou généré automatiquement). La solution à une requête est reconnue sur la base de la similitude entre leurs modèles ayant une certaine sémantique. Avec le niveau d'implication humaine dans la conception d'un tel ensemble de modèles, l'approche montre une grande précision. Il remplace le traitement sophistiqué impliqué dans d'autres méthodes concurrentes, mais son inconvénient est que des règles doivent être écrites pour différents domaines en fonction de la construction et des exigences de différents domaines.

- **Approche basée sur un modèle :**

  Une méthode basée sur un modèle utilise des modèles préformatés pour les requêtes. L'attention de cette méthode est davantage sur l'illustration que sur l'explication des requêtes et des solutions. L'ensemble de modèles est conçu pour comprendre le nombre idéal

de modèles qui assure une couverture adéquate de l'espace du problème.

- **Approche hybride :**

  Différents travaux ont tenté de couvrir les inconvénients de différentes approches en construisant un système plus large qui abrite la combinaison de certains des approches mentionnées. Dans certains de ces systèmes, la réponse est décidée par un système de vote tandis que d'autres attribuent des décisions aux approches qui correspondent le mieux au type de question posée.

5. Chatbot :

Le chatbot est un terme anglais composé de deux mots : “chat” qui signifie conversation, et de “bot”, la moitié du mot “robot”. Il s’agit donc d’un robot conversationnel ou agent conversationnel

Un chatbot est une application d'intelligence artificielle (IA) qui peut imiter une vraie conversation avec un utilisateur dans sa langue naturelle. Il est capable d'interagir en langage naturel et en temps réel, répondre aux questions, proposer des solutions et services adaptés en fonction des requêtes.

Les chatbots permettent la communication via texte ou audio sur des sites Web, des applications de messagerie, des applications mobiles ou par téléphone. [7] [8]

5.1. La classification des Chatbots

La figure qui suit, résume les différentes classes de chatbots :

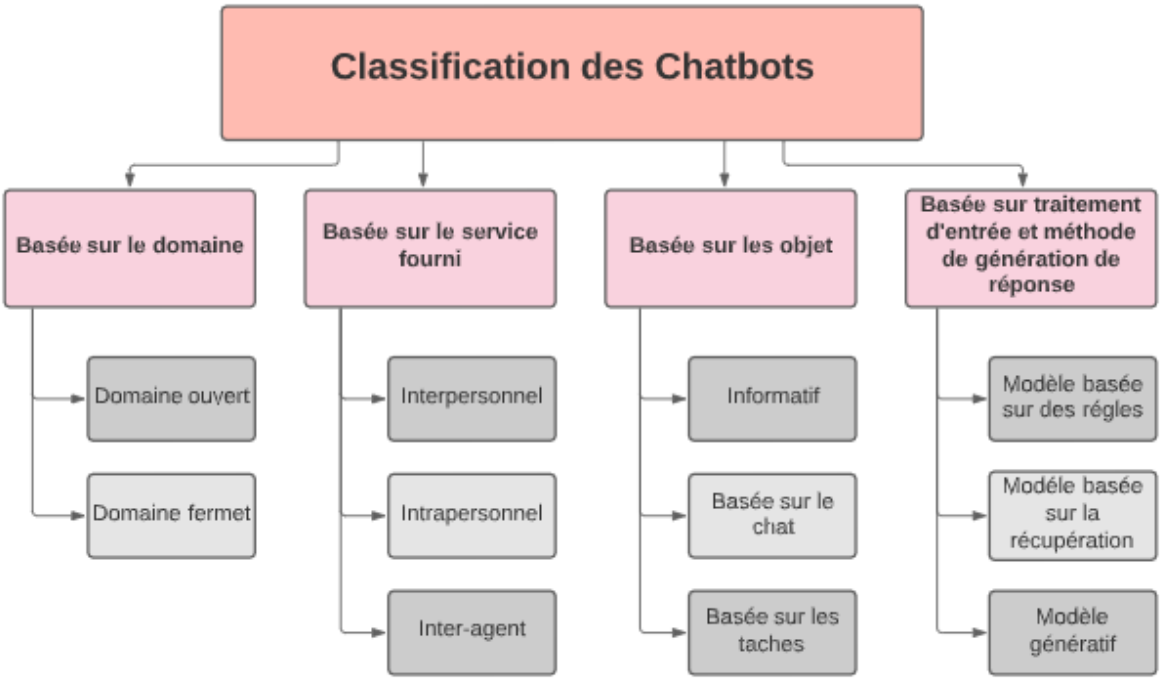


Figure 4: Classification des Chatbots.

Les chatbots peuvent être classés à l'aide de différents paramètres : [9]

5.1.1. Classification basée sur le domaine de connaissances

Les chatbots sont classés en fonction des connaissances auxquelles ils accèdent ou de la quantité de données sur laquelle ils sont formés :

- **Domaine ouvert :** Les chatbots peuvent parler sujets généraux et répondre de manière appropriée.
- **Domaine fermé :** Les chatbots se concentrent sur un domaine de connaissances particulier et peuvent ne pas répondre à d'autres questions.

5.1.2. Classification basée sur le service fourni

Cette classification prend en compte la proximité sentimentale du chatbot avec l'utilisateur, la quantité d'interaction intime qui a lieu et dépend également de la tâche que le chatbot exécute.

- **Interpersonnel :**

Les chatbots relèvent du domaine de la communication et fournissent des services tels que la réservation de restaurant et la réservation de vol Ce ne sont pas des compagnons de l'utilisateur, mais ils obtiennent des informations et les transmettent à l'utilisateur. Ils peuvent avoir une personnalité, être amicaux et se souviendront probablement des informations sur l'utilisateur, mais ils ne sont pas obligés ou attendus de le faire.

- **Intrapersonnel :**

  Les chatbots existent dans le domaine personnel de l'utilisateur, tels que les applications de chat comme Messenger, Slack et WhatsApp. Ils sont les compagnons de l'utilisateur et comprennent l'utilisateur comme un humain.

- **Inter-agent :**

  Les chatbots deviennent omniprésents alors que tous les chatbots nécessiteront des possibilités de communication inter-chatbot. Le besoin de protocoles de communication inter-chatbot est déjà apparu. L'intégration Alexa-Cortana est un exemple de communication inter-agent.

5.1.3. Classification basée sur les objectifs

Les chatbots sont classés en fonction de l'objectif principal qu'ils visent à atteindre :

- **Informatif :**

  Les chatbots sont conçus pour fournir à l'utilisateur des informations préalablement stockées ou disponibles à partir d'une source fixe, comme les chatbots FAQ.

- **Basé sur le chat / conversationnel :**

  Les chatbots parlent à l'utilisateur, comme un autre être humain, et leur objectif est de répondre correctement à la phrase qui leur a été donnée.

- **Basé sur les tâches :**

Les chatbots effectuent une tâche spécifique comme réserver un vol ou aider quelqu'un. Ces chatbots sont intelligents dans le cadre de la demande d'informations et de la compréhension de l'entrée de l'utilisateur. Le robot de réservation de restaurant est un exemple de chatbot basé sur des tâches.

5.1.4. Classification basée sur les entrées/ sorties :

Les systèmes vraiment intelligents génèrent des réponses et utilisent la compréhension du langage naturel pour comprendre la requête. Ces systèmes sont utilisés lorsque le domaine est étroit et que de nombreuses données sont disponibles pour entraîner un système.

Ils prennent en compte la méthode de traitement des entrées et de génération des réponses. Trois modèles sont utilisés pour produire les réponses appropriées :

- **Modèle basé sur des règles :**

  Les chatbots modèles sont le type d'architecture avec lequel la plupart des premiers chatbots ont été construits, comme de nombreux chatbots en ligne. Ils choisissent la réponse du système sur la base d'un ensemble prédéfini de règles fixes, basé sur la reconnaissance de la forme lexicale du texte d'entrée sans créer de nouvelles réponses textuelles. Les connaissances utilisées dans le chatbot sont codées manuellement et organisées et présentées avec des modèles de conversation. Une base de données de règles plus complète permet au chatbot de répondre à plus de types d'entrées utilisateur. Cependant, ce type de modèle n'est pas robuste aux fautes d'orthographe et de grammaire dans la saisie utilisateur.

- **Modèle basé sur la récupération :**

  C’est un modèle un peu différent de celui basé sur des règles, il offre plus de flexibilité car il interroge et analyse les ressources disponibles à l'aide d'API. Un chatbot basé sur l'extraction récupère certains candidats de réponse à partir d'un index avant d'appliquer l'approche de correspondance à la sélection de réponse.

- **Modèle génératif :**

  Génère des réponses d'une meilleure manière que les autres modèles, en fonction des messages des utilisateurs actuels et précédents. Ces chatbots ressemblent davantage à des humains et utiliser des algorithmes d'apprentissage automatique et des techniques d'apprentissage en profondeur. Cependant, il y a des difficultés à les construire et à les former.

Une autre classification pour les chatbots considère la quantité de « aide humaine » dans leurs composants. Assistée par l'homme Les chatbots utilisent le calcul humain dans au moins un élément du chatbot. Les collaborateurs, les pigistes ou les employés à plein temps peuvent incarner leur intelligence dans la logique des chatbots pour combler les lacunes causées par les limitations des chatbots entièrement automatisés. Bien que le calcul humain, comparé aux algorithmes basés sur des règles et à l'apprentissage automatique, offre plus de flexibilité et de robustesse, il ne peut pas traiter une information donnée aussi rapidement qu'une machine, ce qui rend difficile la mise à l'échelle des demandes des utilisateurs.

Les chatbots peuvent également être classés en fonction des permissions fournies par leur plate-forme de développement. Les plates-formes de développement peuvent être open-source, comme RASA, ou peuvent être de code propriétaire comme les plates-formes de développement généralement proposées par de grandes entreprises telles que Google ou IBM. Plateformes open-source permettent au concepteur de chatbot d'intervenir dans la plupart des aspects de la mise en œuvre. Plates-formes fermées, agissent généralement comme des boîtes noires, ce qui peut être un désavantage significatif selon les exigences du projet. Cependant, l'accès aux technologies de pointe peut être considéré comme plus immédiat pour les grandes entreprises. De plus, on peut supposer que les chatbots développés à partir des plates-formes de grandes entreprises peuvent bénéficier d'une grande quantité de données que ces entreprises collectent.

6. Conclusion

Les chatbots et les systèmes de questions – réponses (QA) sont des domaines du traitement automatique de langues qui ont l'air d'être identique ou similaire mais ils ne le sont pas, car ils suivent des approches, des méthodes, des algorithmes et des modèles complètement différents. Sauf l'entrée et la sortie qui est un texte.

Chapitre II :

Entrepôts de données

1. Introduction

Dans le présent chapitre, nous définissons les systèmes d'information décisionnelle ainsi que l'informatique décisionnelle. Nous abordons part la suite, la chaine décisionnelle qui se constitue de la collecte d'information, intégration, organisation et la restitution. Nous parlerons aussi d'entrepôts de données leurs types, composants ainsi que leurs caractéristiques. Ensuite, nous parlons de la modélisation multidimensionnelle et ses différents modèles de représentation des données, nous détaillerons OLAP, sa structure, son architecture, et le langage MDX pour l'interrogation de l'OLAP.

2. Système d'information décisionnel

D'après Alain Fernandez un système d'information décisionnel est défini comme étant : un ensemble d'outils technologiques, méthodiquement assemblés, et déployés en parfaite cohérence avec la stratégie d'entreprise préalablement élaborée. Permet de délivrer les informations pertinentes à chaque manager afin qu'il puisse prendre le plus efficacement possible les meilleures décisions selon son contexte d'action, ses prérogatives et ses objectifs tactiques et stratégiques. [10]

Nous pouvons également définir le SID comme étant un système d'information permettant de mieux maîtriser les évènements antérieurs et appréhender le futur non pas par des intuitions et données non maîtrisées mais par des informations justifiables et cohérentes afin d'apporter un soutien dans le processus de prise de décision.

3. Informatique décisionnelle

Il existe une panoplie de définitions pour l'informatique décisionnelle (ou Business Intelligence (BI) en Anglais), parmi : l'informatique décisionnelle est un terme générique qui inclut les applications, l'infrastructure et les outils, ainsi que les meilleures pratiques qui permettent

l'accès et l'analyse de l'information pour améliorer et optimiser les décisions et les performances. [10]

Nous pouvons définir l'informatique décisionnelle comme étant l'ensemble des outils informatiques permettant de supporter un SID à travers la consolidation d'immense volume de données de l'entreprise réparties dans plusieurs silos applicatifs des systèmes d'information opérationnels, de les analyser et de faire comprendre ce qu'elles expriment afin d'en dégager les informations qualitatives sur lesquelles une décision sera fondée. Toutefois elle présente un système de support et non pas de remplacement du décideur en combinant le traitement de l'information et le jugement humain afin d'avoir un aperçu global de l'état de l'entreprise, d'améliorer la qualité et l'efficacité de la prise de décision d'une part et l'atteinte les objectifs de l'entreprise d'autre part.

4. Chaîne décisionnelle [10]

Une chaîne décisionnelle est mise en place afin de rendre accessible, de mettre en forme et de présenter les informations clés pour faciliter la prise de décision [10]:

Figure 5: modélisation de la chaine décisionnelle.

4.1. Collecte et alimentation

La collecte des données se fait depuis plusieurs sources hétérogènes internes par rapport à une entreprise (bases de données client, données de production, applications métiers, ect.) ou externes (bases de données professionnelles, informations économiques et tous types de données provenant de l'internet). Avant d'être stocker et exploiter, Ces données doivent passer par une famille d'outils dénommée « ETL ».

La figure ci-dessous présente le principe de l'ETL et ses étapes.

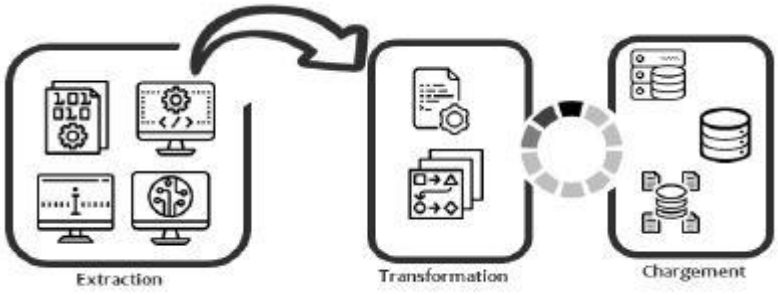


Figure 6: Les étapes du processus ETL.

4.1.1. Principe ETL

L'ETL (Extraction, Transformation and Loading) est un ensemble de fonctionnalités combinées dans une seule solution, pour « extraire » des données depuis un grand nombre de bases de données, applications et systèmes, les « transformer » en fonction des besoins et les « charger » dans une autre base de données pour les analyser, ou bien les envoyer à un autre système opérationnel dans le cadre d'un processus métier.

4.2.L'intégration

Cette deuxième étape est l'intégration des données. Une fois les données centralisées par un outil d'ETL, celles-ci doivent être structurées au sein de l'entrepôt de données. Cette étape est toujours faite par un ETL grâce à un connecteur permettant l'écriture dans le datawarehouse. L'intégration est en fait un pré-traitement ayant pour but de faciliter l'accès aux données centralisées aux outils d'analyse.

Lors de cette étape les données sont transformées et filtrées en vue du maintien de la cohérence d'ensemble (les valeurs acceptées par les filtres de l'outil d'ETL de la fonction de collecte mais qui peuvent introduire des incohérences dans les données centralisées sont soit rejetées, soit intégrées après une phase d'adaptation)

Enfin, c'est aussi durant cette étape que sont effectués les éventuels calculs et agrégations communs à l'ensemble du SID.

4.3.Organisation

Cette étape de diffusion met les données à la disposition des utilisateurs. Elle permet la gestion de droits d'accès et respecte donc des schémas correspondant au profil ou au métier de chacun. Ainsi l'accès direct à l'entrepôt de données n'est pas autorisé. En effet ce genre de pratique ne correspond généralement pas aux besoins des décideurs ou analystes.

L'objectif principal de l'étape est de segmenter les données collectées en contextes qui soient cohérents, simples à utiliser et qui correspondent à une activité décisionnelle particulière (par exemple aux besoins d'un service particulier). Chaque contexte peut correspondre à un datamart, bien que le stockage physique ne soit pas sujet à des règles particulières.

Généralement un contexte d'organisation est multidimensionnel : il est modélisable sous la forme d'un hypercube et peut donc être mis à disposition via un outil OLAP. [11]

4.4.Restitution

La dernière phase concerne la restitution des résultats, on distingue à ce niveau plusieurs types d'outils différents : Les outils de reporting et de requêtes, Les outils d'analyse, La phase de Datamining.

4.4.1. Les outils de reporting et de requêtes

Permettent la mise à disposition de rapports périodiques, pré-formatés et paramétrables par les opérationnels. Ils offrent une couche d'abstraction

orientée métier pour faciliter la création de rapports par les utilisateurs eux-mêmes en interrogeant le datawarehouse grâce à des analyses croisées. [12]

Les rapports sont classés en deux types :

- **Reporting de masse :**

  Le reporting de masse permet de créer à l'avance des modèles de rapport susceptibles d'être souvent demandé par les utilisateurs, c'est ce qu'on appelle les rapports statiques.

- **Reporting ad hoc :**

  L'utilisateur aura accès à des vues métiers conçus en fonction de ses besoins, lui permettront de choisir facilement l'information qu'il souhaite. Les vues font la passerelle entre les données stockées et les besoins de l'utilisateur.

  Ils permettent également la production de tableaux de bord avec des indicateurs de haut niveau pour les managers, synthétisant différents critères de performance.

  Un tableau de bord est un outil de pilotage qui permet de quantifier l'activité d'entreprise à travers la visualisation, suivi et l'exploitation facile des données.

4.4.2. Les outils d'analyse OLAP

Permettent de traiter des données et de les afficher sous forme de cubes multidimensionnels et de naviguer dans les différentes dimensions. Cet agencement des données permet d'obtenir immédiatement plusieurs représentations d'un même résultat, en une seule requête sous une approche descendante des niveaux agrégés vers les niveaux détaillés (Drill-down, Roll-up).

4.4.3. Les outils de Datamining

Offrent une analyse plus poussée des données historiées, permettant de découvrir des connaissances cachées dans les données comme la détection de corrélations et de tendances, l'établissement de typologies et de

segmentations ou encore des prévisions. Le Datamining est basé sur des algorithmes statistiques et mathématiques, et sur des hypothèses métier.

5. Entrepôt de données (DW)

D'après Bill Inmon, un entrepôt de données est une collection de données orientées sujet, intégrées, non volatiles et historiées, organisées pour le support d'un processus d'aide à la décision.

Oracle définit l'entrepôt de donnée comme étant un référentiel structuré de données d'entreprise orientée sujet, évolutive dans le temps et historique, utilisé pour l'extraction de l'information et l'aide à la décision. L'entrepôt de données stocke les données atomiques et agrégées. [10]

Nous pouvons définir l'entrepôt de donnée comme étant une base de données qui sert à conserver les données collectées au sien de l'organisme afin de faciliter l'analyse des données et le Reporting et la prise de décision.

5.1.Caractéristiques d'un entrepôt de données

Les différentes caractéristiques de l'entrepôt de données sont [10] :

- **Orienté sujet :** Une organisation par thème est appliquée en assemblant le maximum d'informations par rapport à un sujet contrairement aux bases de données relationnelles qui regroupent les informations par rapport à une application.
- **Intégré :** Les données proviennent de sources hétérogènes utilisant chacune un type de format. Elles sont intégrées avant d'être proposées à l'utilisation.
- **Non volatile :** Les données ne disparaissent pas et ne changent pas au fil du temps. Les données de l'entrepôt sont en lecture seule, chargées pour la première fois, puis actualisées régulièrement afin de garder la traçabilité de l'information.

- **Historié :** Garder l'historique des informations stockées, ce qui permet de prendre en charge les tendances, les prévisions et le suivi de l'évolution de l'information dans le temps.

6. La modélisation multidimensionnelle

Le modèle Multidimensionnel (MD) représente les données décisionnelles dans un espace à n dimensions, communément appelé hypercube ou cube de données, il se compose de faits contenant les mesures à analyser et de dimensions et les niveaux de hiérarchie. [13]

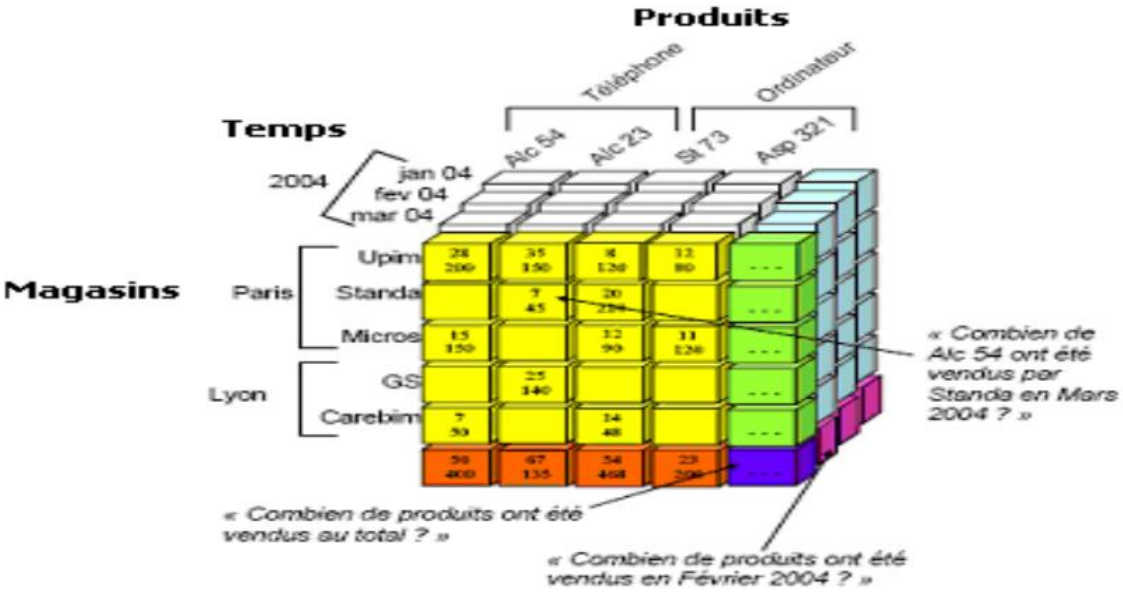


Figure 7: Modèle multidimensionnel

6.1.Modèles de représentation des données

Ci-dessous, nous introduisons quelques concepts sur les modèles de représentation de données [10] [13]**.**

6.1.1. Concepts fondamentaux

Les concepts fondamentaux peuvent être résumés dans ce quisuit :

- **Fait** : Un fait représente un sujet d'analyse composé d'un ensemble de mesures correspondant aux informations de l'activité analysée.
- **Dimension :** Une dimension représente un axe d'analyse en fonction duquel peuvent être observées les sujets analysés (les faits).
- **Granularité :** La granularité est définie comme le niveau de détail maintenu par l'entrepôt. Plus le niveau de détail est élevé, plus le niveau de granularité est fin.
- **Hiérarchie :** Les hiérarchies sont des regroupements naturels au sein d'une dimension et permettent d'agréger les données à différents niveaux. Exemple : la hiérarchie suivante décline la dimension temps avec des niveaux différents : jours -> mois-> Années.

6.1.2. Modélisation conceptuelle

Les schémas de modélisation conceptuelle utilisés lors de la conception d'un entrepôt de données sont :

- **Modèle en étoile**

Ce modèle se présente comme une étoile dont le centre n'est autre que la table des faits et les branches sont les tables de dimension.

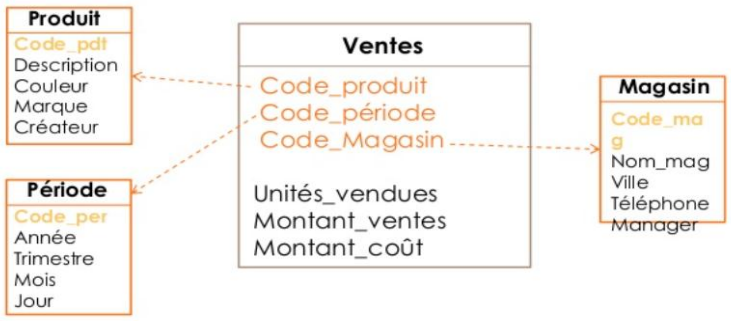


Figure 8: Modélisation en étoile

- **Modèle en flocon**

Le principe est le même que pour le modèle en étoile, mais en plus les dimensions sont décomposées.

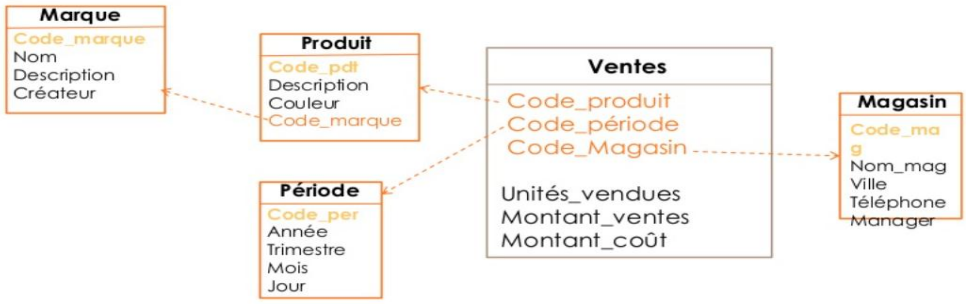


Figure 9:Modélisation en flacon *[14]*

- **Modèle en constellation**

Il est encore basé sur le modèle en étoile. Mais on rassemble plusieurs tables des faits qui utilisent sur les mêmes dimensions.

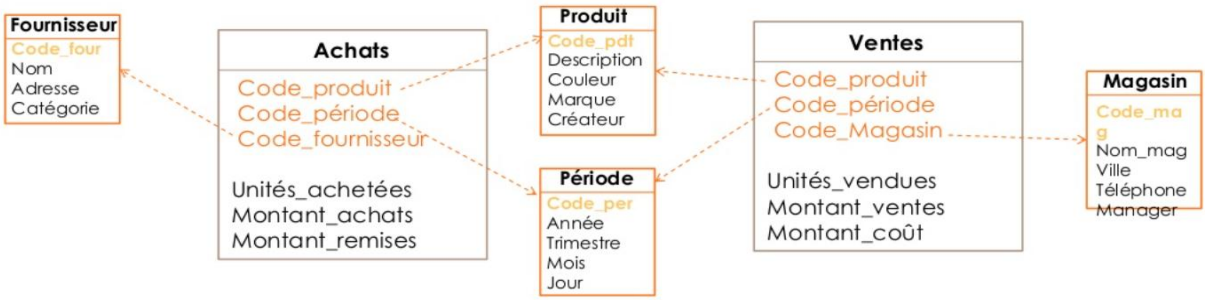


Figure 10: Modèle en constellation *[14]*

6.1.3. Modélisation Logique

Pour représenter une base de données multidimensionnelle sur le plan logique, les modèles ROLAP, MOLAP et HOLAP sont utilisées :

- **OLAP Relationnel (ROLAP)**

Les données sont stockées dans une base de données relationnelle et un moteur OLAP permet de simuler le fonctionnement d'un cube. Cela permet une facilité dans la mise à jour des données.

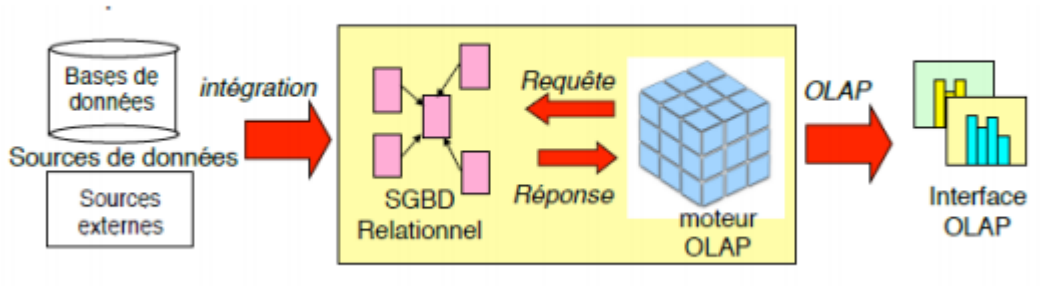


Figure 11: Architecture ROLAP

- **OLAP Multidimensionnel (MOLAP)**

Le Multidimensionnel OLAP consiste à utiliser un système multidimensionnel pur, qui gère des structures multidimensionnelles natives. Elles utilisent des tableaux à n dimensions. L'accès aux données se fait directement dans le cube

Figure 12: Architecture MOLAP

- **OLAP Hybrid (HOLAP)**

Est un hybride entre ROLAP et MOLAP. Les parties tables de faits et tables de dimensions sont stockées dans une base relationnelle standard tandis que le reste des données (les calculs) sont stockées dans une base multidimensionnelle.

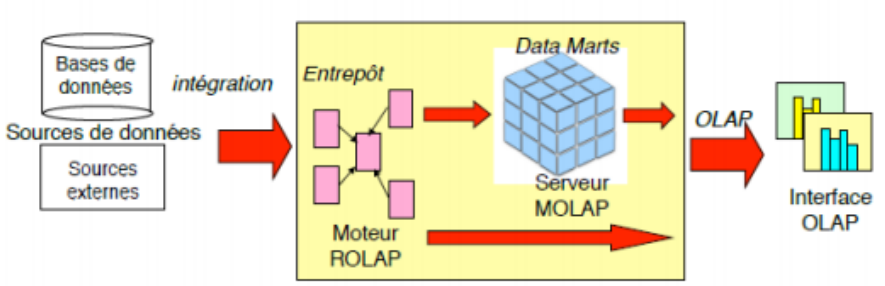


Figure 13: Architecture HOLAP

7. OLAP

OLAP (*acronyme de Online Analytical Processing*). Il 'agit d'une catégorie de logiciels axés sur l'exploration et l'analyse rapide des données selon une approche multidimensionnelle à plusieurs niveaux d'agrégation

7.1.Structure OLAP

Les différents concepts liés à la structure d'OLAP se résument dans ce qui suit [15] **:**

- **Cube :** Le cube OLAP est une structure de données optimisée pour une analyse de données très rapide. Il se compose de faits numériques appelés mesures, classés par dimensions. OLAP Cube est également appelé hypercube. Le cube peut stocker et analyser des données multidimensionnelles de manière logique et ordonnée.
- **Membre :** Un élément de hiérarchie qui représente une ou plusieurs instances de données. Il existe deux catégories de membres : - Unique : un seul membre sans répétition. - Pas unique : un membre qui peut répéter.
- **Les mesures :** Les mesures sont les valeurs de base du cube qui sont prétraitées, regroupées et analysées.
- **Membre calculé :** Un élément de dimension. Sa valeur est calculée au moment de l'exécution à l'aide de l'expression. Les valeurs

calculées des composants peuvent être dérivées des valeurs d'autres éléments.

- **Dimension** : Un ensemble d'une ou plusieurs hiérarchies organisées de niveaux dans un cube. Il est compréhensible pour l'utilisateur et utilisé comme base pour l'analyse des données.
- **Hiérarchie :** Une arborescence logique qui organise les éléments de dimension de manière parentale.
- **Niveau :** Le mode d'organisation des données à l'intérieur de la hiérarchie à des niveaux de détail de plus en plus élevés.

7.2.Architecture d'OLAP

L'architecture OLAP se compose en trois services [13] :

- **Base de données**

  Elle doit supporter les données agrégées ou résumées qui proviennent d'un entrepôt de données possédant une structure multidimensionnelle (SGDB)

- **Serveur OLAP**

  Le serveur OLAP permet d'effectuer une analyse de données conforme au paradigme multidimensionnel, avec des temps de réponse optimisés. Un serveur OLAP fournit aux utilisateurs une représentation multidimensionnelle des données sous forme d'un ensemble d'hyper-cubes et implémente un ensemble d'opérateurs OLAP (Roll-up, Drill-down, etc.) qui permettent d'explorer ces hyper cubes.

- **Client OLAP**

  Cette couche définit une série d'interfaces utilisateurs intuitives pour l'exploration interactive et multidimensionnelle des données. Ces interfaces permettent de déclencher les opérateurs OLAP et présentent l'information en utilisant différents types d'affichages

interactifs : tableaux croisés dynamiques, histogrammes et diagrammes statistiques

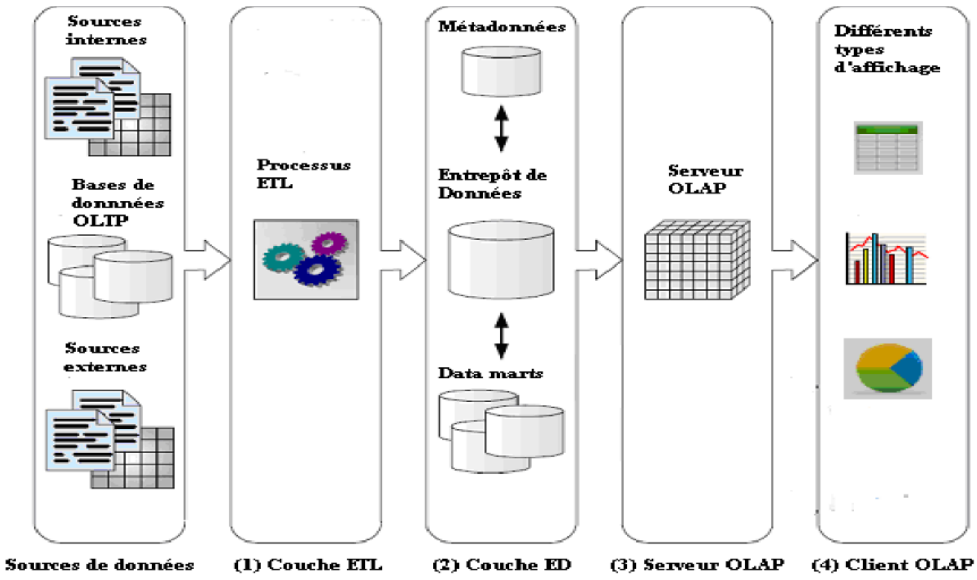


Figure 14 : Architecture typique d'un système OLAP.

7.3.Opérations de manipulation OLAP :

Les opérations analytiques de base suivantes sont utilisées pour l'analyse multidimensionnelle [16].

- **Drill up / Drill Down** :

Permet d'aller vers les informations détaillées dans une hiérarchie ou au contraire de remonter d'un niveau de granularité. Il s'agit donc de « zoomer ou de dézoomer » sur une dimension.

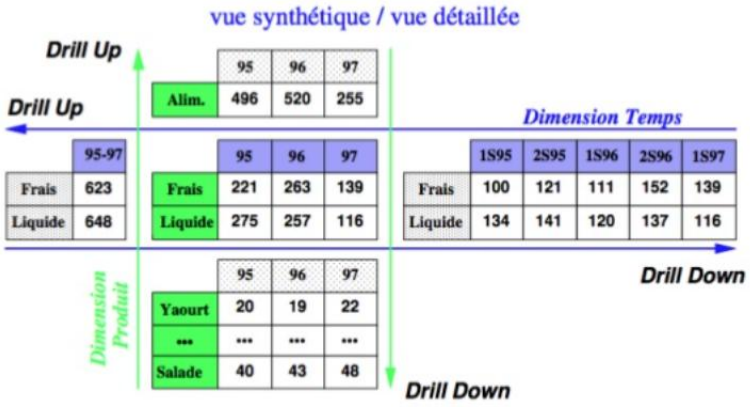


Figure 15: Opération de manipulation : Drill up/Drill Down. *[14]*

- **Rotation (pivot) :**

Consiste à effectuer une rotation du cube multidimensionnel afin de présenter une face différente. Il s'agit donc de modifier une dimension de lecture.

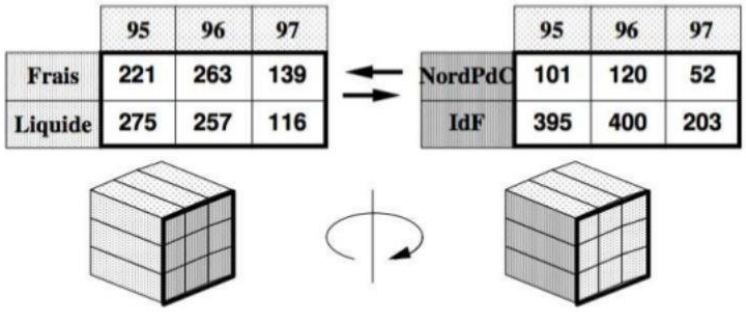
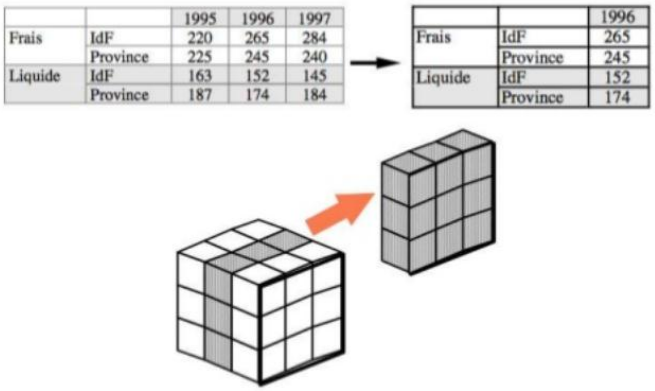


Figure 16: Opération de manipulation : Rotation *[14]***.**

- **Opérations de sélection (Slice)** :

Consiste à ne travailler que sur une tranche du cube. Une des dimensions est alors réduite à une seule valeur.



		1995	1996	1997
Frais	IdF	220	265	284
	Province	225	245	240
Liquide	IdF	163	152	145
	Province	187	174	184

		1996
Frais	IdF	265
	Province	245
Liquide	IdF	152
	Province	174

Figure 17 : Opération de manipulation : Sélection *[14]***.**

- **Opérations de projection (Dice) :**

Consiste à ne travailler que sur un sous-cube. On s'intéressera alors seulement à une partie des données.

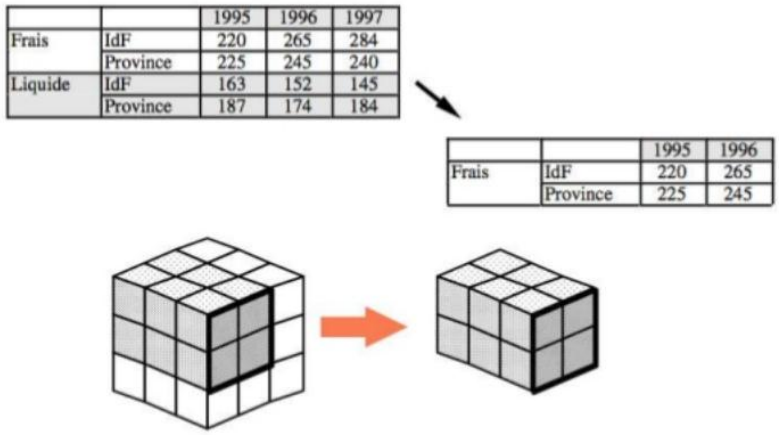


Figure 18: Opération de manipulation Projection *[14]*.

7.4. Comparaison entre OLAP et OLTP

SGBD et datawarehouse ont des objectifs différents. Ils stockent les données de manière différentes et font l'objet de requêtes différentes. Ils sont ainsi basés sur deux systèmes différents [17] : OLTP et OLAP

7.4.1. OLTP

OLTP (On Line Transaction Processing) est le modèle utilisé par les SGBD. Le mode de travail est transactionnel. L'objectif est de pouvoir insérer, modifier et interroger rapidement et en sécurité la base. Ces actions doivent pourvoir être effectuées très rapidement par de nombreux utilisateurs simultanément.

Chaque transaction travail sur de faibles quantités d'informations, et toujours sur les versions les plus récentes des données.

7.4.2. Comparaison

Voici un tableau récapitulatif des différences entre OLTP et OLAP :

Caractéristiques	OLTP	OLAP
Utilisation	SGBD (base de production)	Datawarehouse
Opération typique	Mise à jour	Analyse
Type d'accès	Lecture écriture	Lecture
Niveau d'analyse	Elémentaire	Global

Caractéristiques	OLTP	OLAP
Quantité d'information échangées	Faible	Importante
Orientation	Ligne	Multidimensionnel
Taille BD	Faible (max qq GB)	Importante (pouvant aller à plusieurs TB).
Ancienneté des données	Récente	Historique

Tableau 1: tableau récapitulatif des différences entre OLTP et OLAP.

L'objectif des bases OLTP est de pouvoir répondre rapidement à des réponses simples, exemple : les ventes du produit X.

Les bases OLAP permettent des requêtes plus complexes : les ventes du produit X par vendeur, région et par mois.

7.5.MDX pour l'interrogation d'OLAP

Pour effectuer des requêtes au sein des cubes OLAP, on utilise le langage MDX ou le langage SQL avec le produit ROLAP.

7.5.1. Définition :

MDX (Multidimensional Expressions) est un langage de requête pour les bases de données OLAP, tout comme SQL est un langage de requête pour les bases de données relationnelles MDX est un langage de requête utilisé pour interagir et effectuer des tâches avec des bases de données multidimensionnelles (également appelées : Cubes OLAP) [18].

Le langage MDX a été développé à l'origine par Microsoft à la fin des années 1990, et a été adopté par de nombreux autres fournisseurs de bases de données multidimensionnelles.

7.5.2. Structure et utilisation de la requête MDX

Une requête MDX de base utilise l'instruction SELECT pour identifier un ensemble de données qui contient un sous-ensemble de données

multidimensionnelles. L'instruction SELECT est composée des clauses suivantes : Clause

- **WITH (facultative) :**

  Permet de calculer des membres calculés ou des ensembles nommés pendant le traitement des clauses SELECT et WHERE. Clause

- **SELECT :**

  Définit les axes de la structure de requête MDX en identifiant les membres de dimension à inclure sur chaque axe. Le nombre de dimensions d'axe d'une instruction MDX SELECT est également déterminé par la clause SELECT. Clause

- **FROM :**

  Nomme le cube interrogé et détermine quelle source de données multidimensionnelle sera utilisée lors de l'extraction des données pour remplir le jeu de résultats de l'instruction MDX SELECT. La clause FROM (dans une requête MDX) ne peut répertorier qu'un seul cube. Les requêtes sont limitées à une seule source de données ou à un cube. Clause

- **WHERE (facultative) :**

  Détermine quelle dimension ou quel membre est utilisé comme dimension de segment (le segment fait généralement référence à l'axe formé par la clause WHERE). Cela limite l'extraction des données à une combinaison de membres de dimension. Toute dimension qui n'apparaît pas sur un axe dans la clause SELECT peut être nommée sur le segment.

7.5.3. Comparaison entre SQL et MDX :

MDX est fait pour naviguer dans les bases multidimensionnelles et pour exécuter des requêtes sur tous leurs objets (dimensions, hiérarchies, niveaux,

membres et cellules) afin d'obtenir (simplement) une représentation sous forme de tableaux croisés [19].

La syntaxe de MDX ressemble à celle de SQL par ses mots clé SELECT, FROM, WHERE, Mais leurs sémantiques sont différentes :

- SQL construit des vues relationnelles.
- MDX construits des vues multidimensionnelles des données.

Analogies entre termes multidimensionnels (MDX) et relationnels (SQL) :

Relationnel (SQL)	Multidimensionnel (MDX)
Table	Cube
Colonne (chaine de caractère ou valeur numérique)	Niveau (Level)
Plusieurs colonnes liées ou une table de dimension	Dimension
Colonne (discrète ou numérique)	Mesure (Measure)
Valeur dans une colonne et une ligne particulière de la table	Membre de dimension (Dimension member)

Tableau 2: Comparaison entre SQL et MDX.

- **Structure générale d'une requête :**

- SQL: SELECT column1, column2, …, columnn FROM table
- MDX: SELECT axis1 ON COLUMNS, axis2 ON ROWS FROM cube

Clause FROM spécifie la source de données :

- En **SQL :** une ou plusieurs tables
- En **MDX :** un cube

Clause SELECT indique les résultats que l'on souhaite récupérer par la requête :

- En **SQL :**
  - Une vue des données en 2 dimensions.
  - Lignes (rows) et colonnes (columns), les lignes ont la même structure définie par les colonnes
- **En MDX :**
  - Nombre quelconque de dimensions pour former les résultats de la requête
  - Terme d'axe pour éviter confusion avec les dimensions du cube
  - Pas de signification particulière pour les rows et les columns, mais il faut définir chaque axe : axe1 définit l'axe horizontal et axe2 définit l'axe vertical

- **Le contenu des axes :**

Un membre (Member) = est un item dans une dimension et correspond à un élément spécifique de donnée :

[Time]. [2012]
[Customers].[All Customers].[Mexico].[Mexico]
[Product].[All Products].[Drink]
Un tuple= une collection de membres de différentes dimensions :
([Time].[2012], [Product].[All Products].[Drink])
(2012, Drink)
(2012, [Customers].[All Customers].[Mexico].[Mexico])
Un set = une collection de tuples :

8. Conclusion

Dans le but de prendre des décisions pertinentes au sein d'un organisme doté d'un système d'information possédant un entrepôt de données géré par

un server OLAP, les données doivent être visibles d'une façon à permettre à l'utilisateur de choisir facilement l'information qu'il souhaite ; soit via les outils de visualisation OLAP qui restent un moyen pas toujours efficace pour obtenir un aperçu détaillé, soit par l'interrogation avec des requête MDX ce qui n'est pas toujours accessible aux non experts. Ceci traduit le besoin a une interface d'interrogation en langage naturel.

Chapitre III :
Conception et modélisation de la solution

1. Introduction

Dans le présent chapitre, nous abordons toutes les étapes de conception de notre solution capitalisant les besoins cernés.

En commençant par les spécifications fonctionnelles et techniques du projet ainsi que l'architecture proposée les concrétisant.

Nous passant ensuite à la modélisation de notre système afin de réduire la complexité de ce dernier lors de l'implémentation et d'organiser la réalisation du projet en définissant les différentes étapes.

2. Solution proposée

2.1. Architecture fonctionnelle

Notre démarche est composée de six étapes qui seront détaillées un peu plus bas :

1) Récupération de la requête en langage naturel.
2) Vérification du contexte de la requête.
3) Extraction des mots clés.
4) Génération de la requête MDX équivalente.
5) Exécution de la requête MDX sur le serveur OLAP
6) Affichage du résultat à l'utilisateur.

La figure suivante (19) résume les différentes étapes que nous avons suivies pour réaliser notre système. En commençant par l'introduction de la requête passant par sa validation et l'extraction des mots clés ensuite la génération de la requête mdx et finissant par l'exécution de la requête générer et l'affichage des résultats.

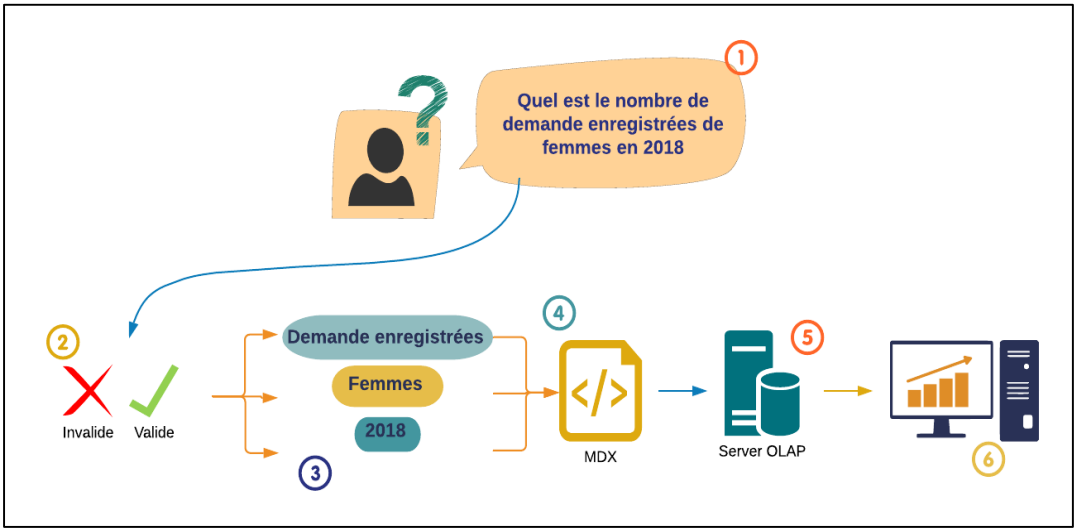


Figure 19 : Représentation du traitement d'un exemple.

2.2. Architecture technique

La Figure ci-dessous (19) décrit l'interaction des différentes parties de notre système entre l'interface utilisateur front-end et la back-end de notre système ce fait la conversion et l'exécution de la requête.

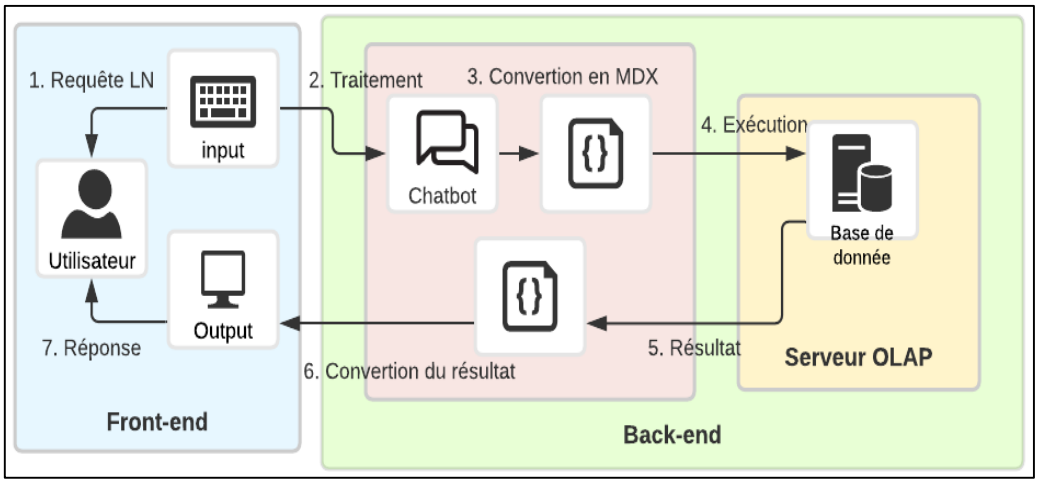


Figure 20: Interaction entre les parties du système.

1. L'utilisateur final saisit une requête en langage naturel.
2. Dans cette étape la requête est traitée à l'aide d'un chatbot afin d'extraire les paramètres nécessaires.
3. Grace aux paramètres récupérés de la requête initiale, une requête MDX adéquate est automatiquement générée.
4. Exécution de la requête MDX générée sur le serveur OLAP de l'ANEM grâce à une chaine de connexion.

5. Récupération des résultats avec la même chaine de connexion.
6. Les résultats récupérés seront convertis en html pour être affichés à l'utilisateur final.
7. L'utilisateur reçoit la réponse voulue.

3. Modélisation

Plusieurs démarches de modélisation sont utilisées. Nous adoptons dans notre travail une approche objet basée sur un outil de modélisation UML.

Pour modéliser notre système, nous allons identifier trois diagrammes :

- Le diagramme de cas d'utilisation qui recenser les principales fonctionnalités de notre système.
- Le diagramme d'activités qui décrit la succession des activités et leurs interactions dans un processus du système.
- Le diagramme de séquences qui permet de visualisé l'interaction des différents éléments de notre système et la chronologie d'échange de message.

3.1. Diagramme de cas d'utilisation

Le diagramme suivant représente le cas d'utilisation de notre architecture (figure21). Il décrit le comportement du système du point de vue utilisateur. En effet, l'utilisateur va introduire en requête en langage naturel, cette requête vas être traites à l'aide d'un chatbot ou vas se faire l'extraction des mots clé. Ensuite le système va faire la conversion de la requête en LN vers une requête MDX grâce à ces mots clés, la requête générer vas être exécuté au sein du server OLAP.

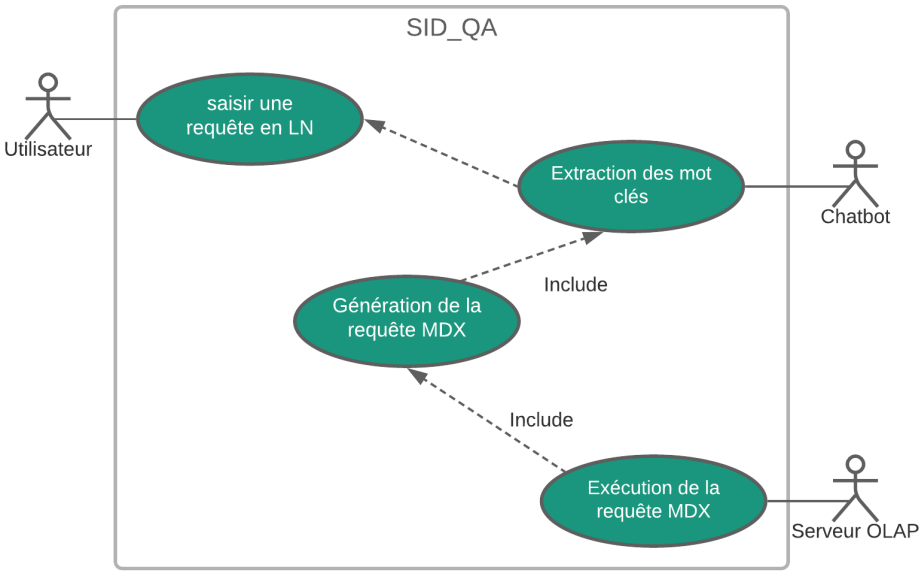


Figure 21 : Diagramme de cas d'utilisation

3.2. Diagramme d'activités

Six étapes sont définies (figure 22) :

- **Première Etape : La saisie d'une requête en langage naturel**

Dans cette étape l'utilisateur introduit une requête en langage naturel qui exprime son besoin en information. Cela se fait grâce à une interface web utilisateur.

- **Deuxième Etape : Vérification du contexte**

Cette étape prend en entrée une phrase qui est la requête en langage naturel afin de la classer comme valide si la phrase est bien formulée pour pouvoir passé à l'étape suivante, ou non si elle est hors contexte définit et demander à l'utilisateur de réintroduire une autre phrase.

Exemple :

- Quel est le nombre de femmes inscrites en 2018. (Valide)
- Quel est le nombre de femmes retraités en 2018. (Non valide)

- **Troisième Etape : Extraction des mots clés**

Après la vérification, vient l'étape d'extraction des mots clés qui est l'étape la plus importante où les paramètres nécessaires à la construction de la requête MDX sont prélevés et classés selon leurs rôles dans la base de données (Dimensions et Mesures) et avec la valeur de chacun.

Exemple :

Dimension : genre Valeur : femmes

- **Quatrième Etape : Générer une requête mdx**

Dans cette étape la requête MDX commence à se former, elle est construite avec les paramètres récupérés dans l'étape précédente et cela en respectant la syntaxe du langage de requête MDX.

- **Cinquième Etape : Exécution de la requête mdx**

La requête MDX générée sera exécutée au sein du serveur OLAP du système d'information décisionnelle.

- **Sixième Etape : Affichage du résultat**

Le résultat émit par le système sera affiché à l'utilisateur.

La Figure suivante (21) représente le diagramme d'activités qui décrit les différentes étapes et qui représente le déclenchement de chaque événement de notre système.

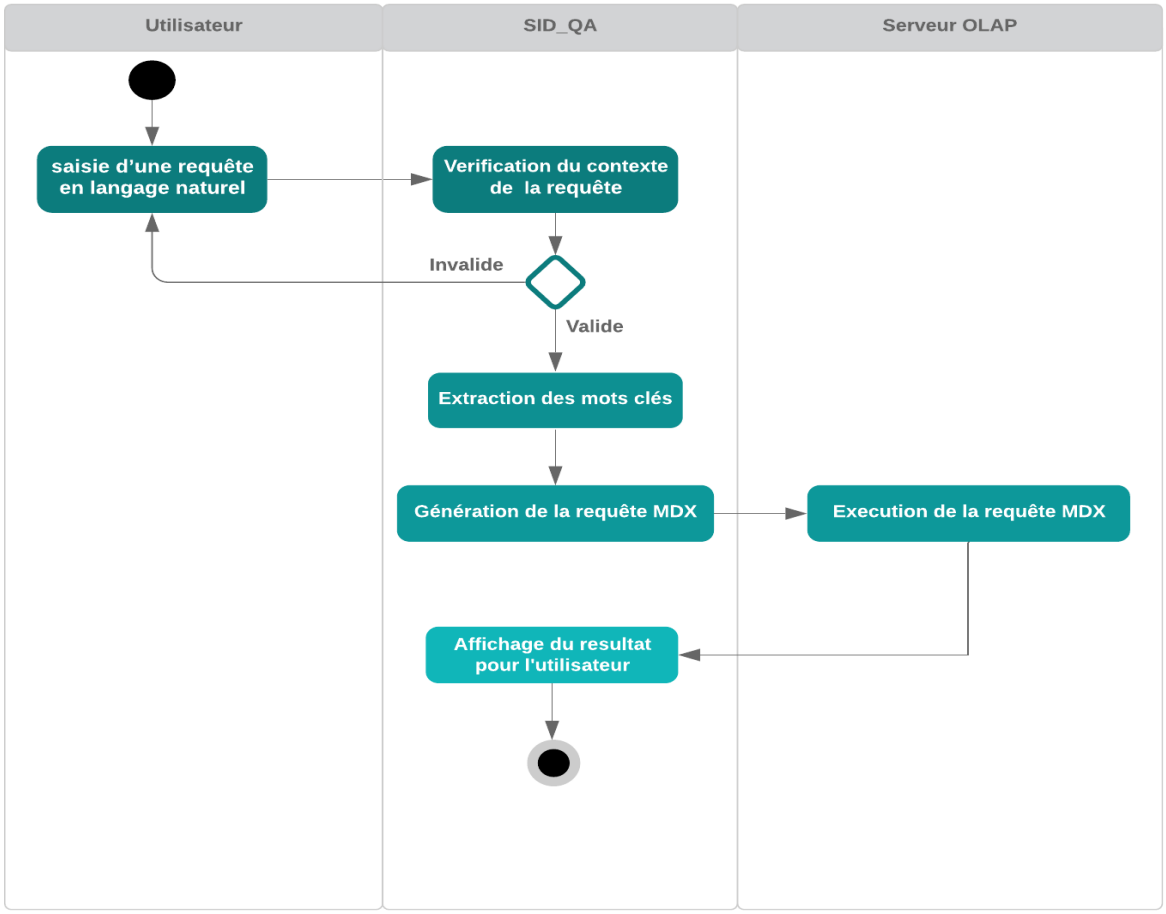


Figure 22: Diagramme d'activités du système.

3.3. Diagramme de séquences

La figure (Figure 23) ci-dessous montre le processus d'interogation du SID en LN.

Il regroupe les interactions des éléments de notre systéme et résume les difirentes étape sité dans le diagramme d'activité dans un odre chronologique permétent de visualiser les fonction de chaque élément.

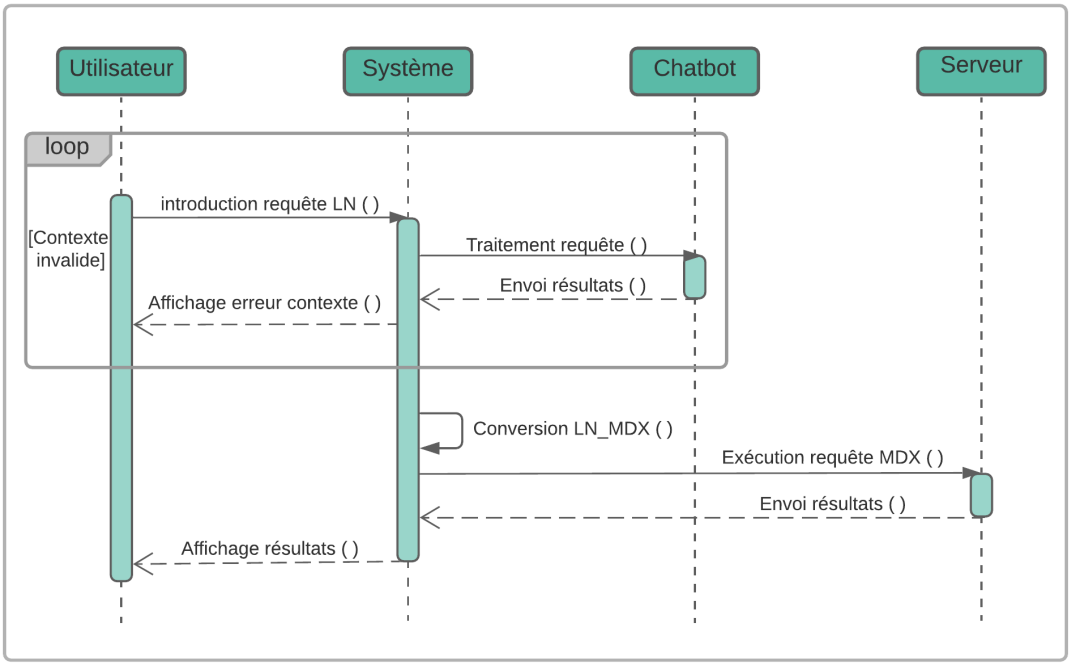


Figure 23: Diagramme de séquences

4. Conclusion

Dans ce chapitre, nous avons présenté l'architecture fonctionnelle est technique de notre solution ainsi que sa modélisation, cela pour faciliter la réalisation de notre système.

Dans le chapitre suivant nous montrerons les étapes, plus en détails, que nous avons suivies pour implémenter et réaliser notre solution.

Chapitre IV :
Implémentation de la solution

1. Introduction

L'objectif de la phase d'implémentation est d'aboutir à un produit exploitable par les utilisateurs. Cette phase consiste à transformer le modèle conceptuel établi précédemment en des composants logiciels formant notre système.

Le choix des outils de développement influe énormément sur le coût en temps de programmation, ainsi que sur la flexibilité du produit à réaliser.

Dans ce chapitre, nous spécifions les outils, langages et techniques que nous avons choisies et mis en œuvre, et nous finirons par présenter un scénario de notre application illustré par des captures d'écrans.

2. Environnement de développement

2.1. Outils

- **Django**

C'est un cadre de développement web open source en Python de haut niveau qui encourage un développement rapide et une conception propre et pragmatique. Développé en 2003 et publié à partir de juillet 2005 Django est gratuit et open source. Il comporte un serveur web léger permettant de développer et tester ses applications en temps réel sans déploiement. [20]

- **Dialogflow**

C'est une plate-forme de compréhension du langage naturel qui facilite la conception et l'intégration d'une interface utilisateur de conversation dans différente applications mobiles, Web, bots, etc. l'API Dialogflow peut aussi être utilisée pour créer des agents pour des scénarios avancés. [21]

- **SQL Server**

C'est un système de gestion de base de données (SGBD) en langage SQL incorporant entre autres un SGBDR (SGBD relationnel) développé et commercialisé par la société Microsoft. Il fonctionne sous les OS Windows et Linux, il est aussi possible de le lancer sur Mac OS via Docker. [22]

- **SQL Server Management Studio (SSMS)**

C'est un environnement intégré pour la gestion de toute infrastructure SQL. Il utilise SSMS pour accéder, configurer, gérer, administrer et développer tous les composants de SQL Server, Azure SQL Database et Azure Synapse Analytics. SSMS fournit un seul utilitaire complet qui combine un large groupe d'outils graphiques avec un certain nombre d'éditeurs de scripts riches pour fournir un accès à SQL Server pour les développeurs et les administrateurs de base de données de tout niveau. [23]

- **ADOMD.NET**

ADOMD.NET est un fournisseur de données Microsoft .NET Framework conçu pour communiquer avec Microsoft SQL Server Analysis Services. ADOMD.NET utilise le protocole XMLA (XML for Analysis) pour communiquer avec les sources de données analytiques en utilisant des connexions TCP/IP ou HTTP pour transmettre et recevoir des demandes et des réponses SOAP conformes à la spécification XML for Analysis. [24]

2.2. Langages de programmations

- **Python**

Python est un langage de programmation open source créé par le programmeur Guido van Rossum en 1991. Il tire son nom de l'émission Monty Python's Flying Circus.

Il s'agit d'un langage de programmation interprété, qui ne nécessite donc pas d'être compilé pour fonctionner

En tant que langage de programmation de haut niveau, Python permet aux programmeurs de se focaliser sur ce qu'ils font plutôt que sur la façon dont ils le font. **[25]**

- **HTML**

Acronyme anglais de HyperText Markup Language, généralement abrégé HTML est un langage de balisage, 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 **[26]**.

- **CSS**

Le terme CSS est l'acronyme anglais de Cascading Style Sheets qui peut se traduire par "feuilles de style en cascade". Le CSS est un langage informatique utilisé sur l'internet pour

mettre en forme les fichiers HTML ou XML. Ainsi, les feuilles de style, aussi appelé les fichiers CSS, comprennent du code qui permet de gérer le design d'une page en HTML **[27]**

3.Mise en œuvre

3.1. Création du chatbot

Le chatbot que nous avons créé est nommé SID_Q&A et les étapes de sa création sont comme suit :

3.1.1. Intentions (Intent)

L'intent dans Dialogflow catégorise l'intention de l'utilisateur dans une conversation. De nombreux intents peuvent être définie. Dialogflow fait correspondre l'expression de l'utilisateur final à la meilleure intention de l'agent. Un intent de base contient des phrases d'entraînement, des actions, des paramètres et des réponses.

Dans notre cas l'intent représente le nom de cube sur lequel se fera la projection de l'information voulue. Comme le montre la figure 22 en rouge l'intent nommé « Analyse de l'orientation » représente le cube sur lequel nous avons effectué nos tests.

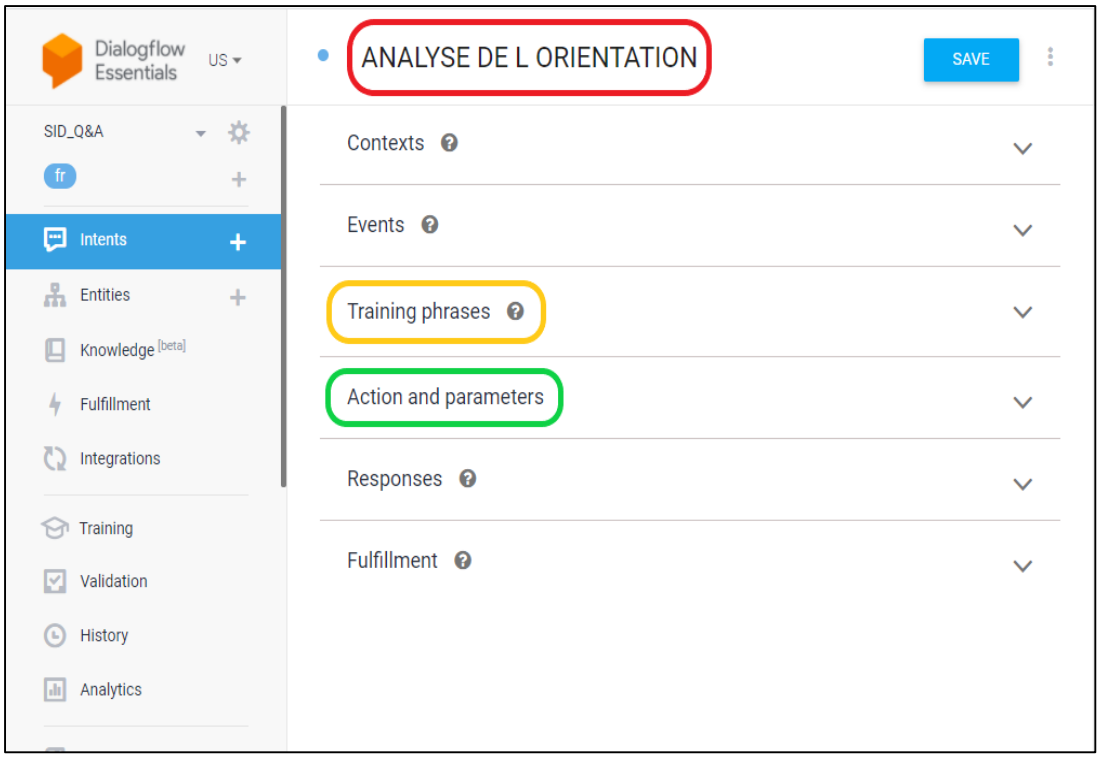


Figure 24: Analyse de l'orientation.

3.1.2. Entités

Chaque paramètre d'intent a un type, appelé type d'entité, qui dicte exactement comment les données d'une expression d'utilisateur final sont extraites.

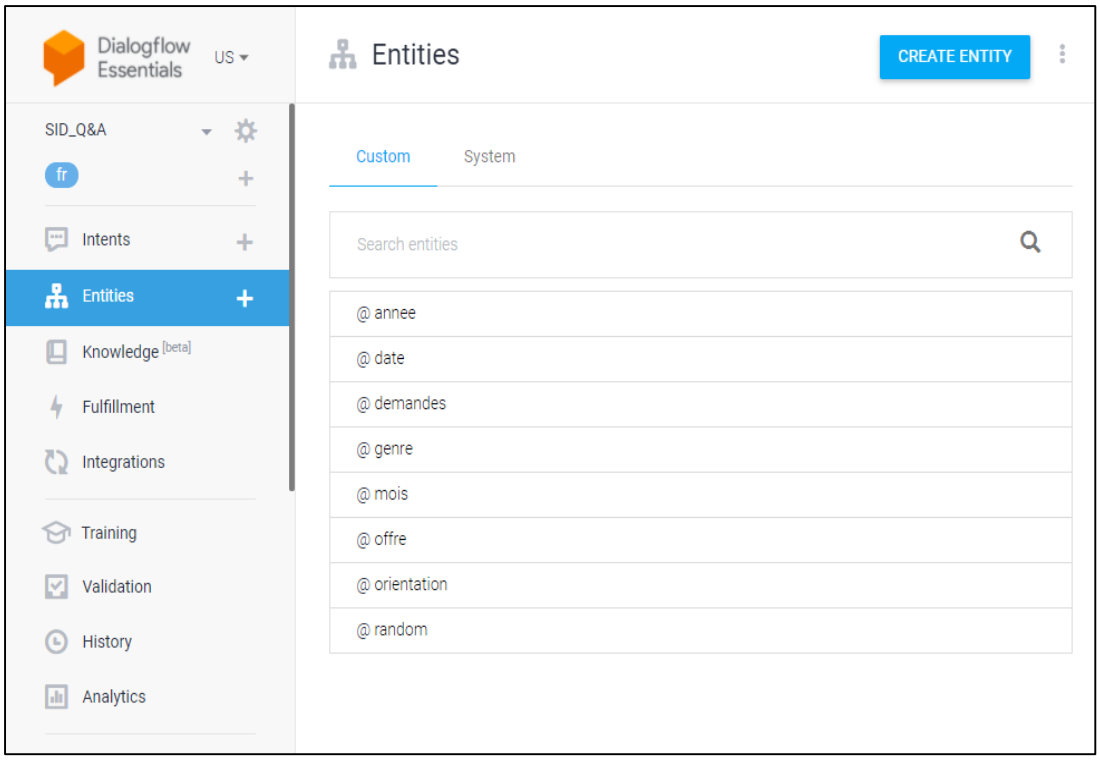


Figure 25: Groupe d'entités

Nous avons défini plusieurs entités (Figure 22) qui correspondent aux informations disponibles dans la base de données. Comme le montre la (Figure 23) nous avons défini une entité « genre » qui contient les genres homme et femme avec leur synonymes.

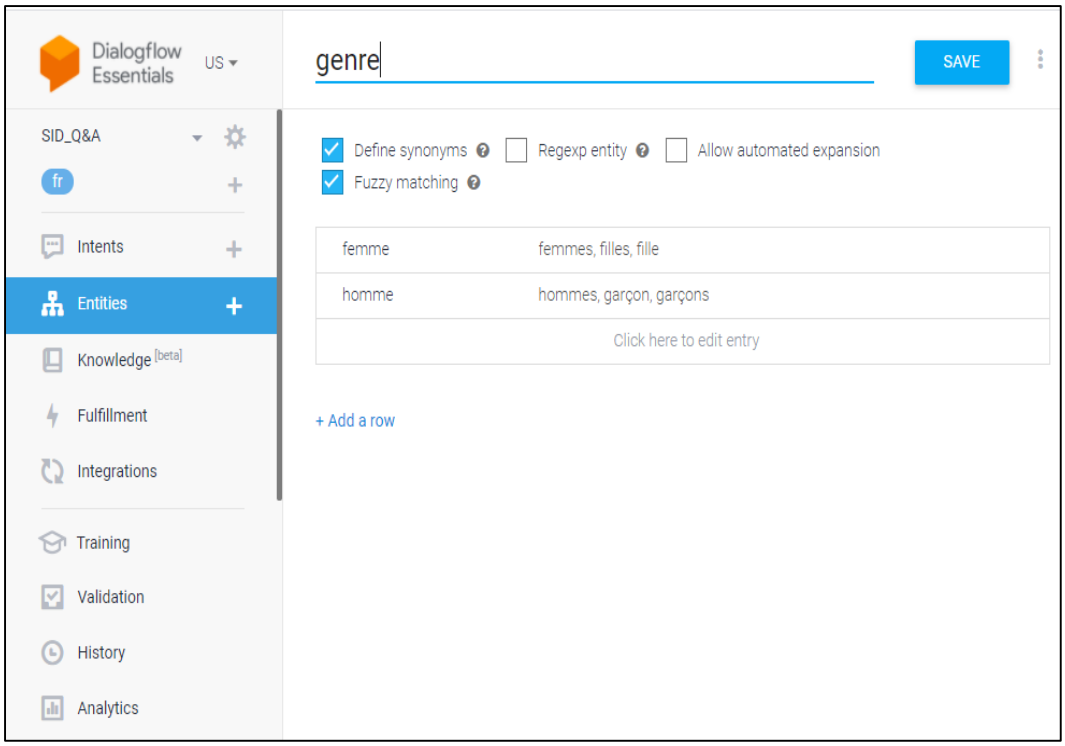


Figure 26: Entité "Genre"

3.1.3. Phrases d'entraînement (Training phrases)

Comme indiqué dans la Figure 22, en jaune les phrases d'entraînement sont des exemples d'expressions de ce que l'utilisateur final pourrait dire et qui corresponde à cette intention. Et grâce à l'apprentissage automatique intégré de Dialogflow cette liste et étendue avec d'autres expressions similaires. La Figure 25 montre un exemple de phrases que nous avons introduit.

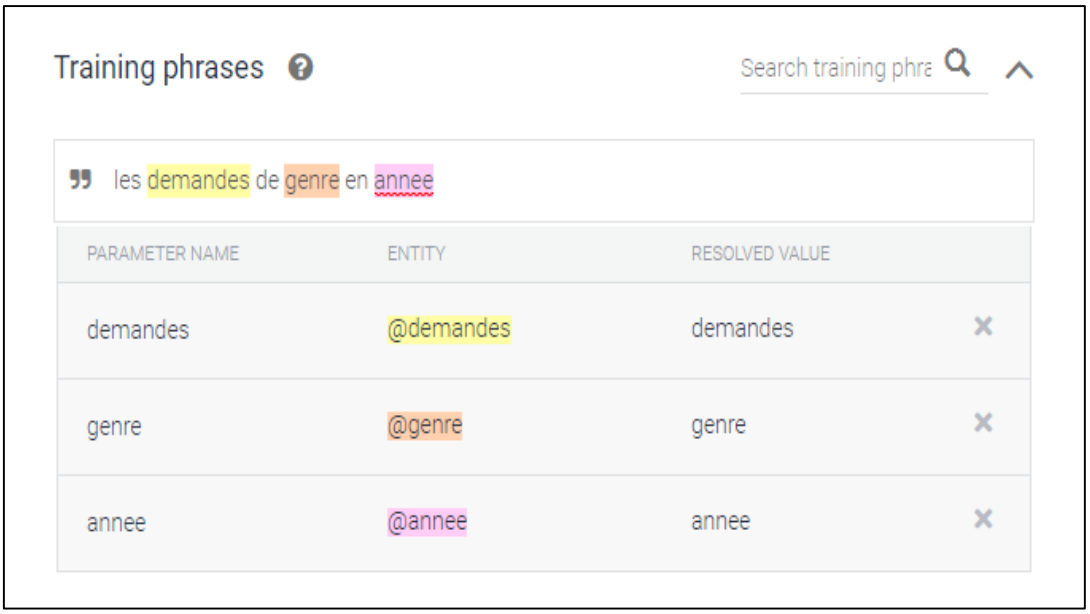


Figure 27 : Exemple de phrase d'entrainement avec paramètres

3.1.4. Paramètres

Ce sont des données structurées qui peuvent être utilisées pour exécuter une logique ou générer des réponses. Chaque paramètre possède un type appelé type d'entité et une valeur qui est définie dans cette entité. Comme le montre la Figure 25 nous avons utilisé trois paramètres dans une phrase et qui sont demandes, genre, et année. Chaque intent a un groupe de paramètre comme le montre la Figure 22 en vert.

3.2. Constitution de la requête MDX

Une requête MDX est composée de plusieurs parties. Sa composition doit être minutieuse car chaque caractère compte. Les étapes que nous avons choisies et suivies sont :

3.2.1. Récupération des paramètres

Comme mentionné dans la création du chatbot la phrase introduite par l'utilisateur contient des paramètres, ces paramètres sont nécessaires à la conception de la requête MDX.

Pour récupérer ces paramètres nous avons intégré dans notre système une API que propose Dialogflow. Ces données sont récupérées sous forme de dictionnaire avec « clé, valeur » où les clés représentent les entités et les valeurs sont celles introduites pour cette même entité.

3.2.2. Classification des mesures et dimensions

Les paramètres que nous avons récupérés représente les mesures et les dimensions regroupées dans le même dictionnaire, nous avons séparé ces données en prennent en compte les informations qui existent dans la base de données. Nous avons mis les noms de mesures dans une liste.

Les dimensions quant à elles sont mises dans un dictionnaire car nous avons besoin du nom de la dimension et de sa valeur si la dimension et sans valeurs elle ne sera pas prise en compte comme nous pouvons avoir une dimension avec plusieurs valeurs. Nous avons réalisé cela de la fonçons suivante :

```
pM = []
pD = {}
for e in parameters.items():
    for x in e[1]:
        if x in liste_meussures:
            pM.append(x)
    if e[0] in liste_dimentions and e[1] != '' and e[1] != []:
        if e[1] != []:
            for x in e[1]:
                pD[e[0]] = x
        else:
            pD[e[0]] = e[1]
```

Figure 28: code de classification des mesures et dimensions

3.2.3. Partie 1 : Sélection sur les mesures

Une fois les mesures ont été mises dans une liste, nous allons parcourir cette liste tout en formant cette partie de la requête qui doit être comme suit :

NON EMPTY {[Measures]. [Nom de la mesure], [Measures]. [Nom de la mesure]…..}

```
def part1(p):
    l = 'NON EMPTY { '
    for e in pM:
        l += '[Measures].[{0}] ,'.format(e)
    l = l[:-1]
    l += ' }'
    return l
```

Figure 29: Fragment de code de la sélection sur les mesures

3.2.4. Partie 2 : Projection sur les dimensions

Pour cette partie nous allons parcourir le dictionnaire contenant les dimensions et leurs valeurs et nous allons projeter les dimensions comme suit :

NON EMPTY ([nom de dimension]. [Valeur]….)

```
def part2(p):
    if pD == {}:
        l = ''
    else:
        l = 'NON EMPTY ( '
        for e in pD.items():
            if type(e[1]) is list:
                for i in e[1]:
                    l += '[{0}].[{1}] ,'.format(str(e[0]), str(i))
            else:
                l += '[{0}].[{1}] ,'.format(str(e[0]), str(e[1]))
        l = l[:-1]
        l += ' )'
    return l
```

Figure 30:lignes de code de la projection sur les dimensions

Tout en notant que chaque dimension peu avoir une ou plusieurs valeurs ou que cette partie peut être vide si nous n'avons pas de projection sur les dimensions.

3.2.5. Finalisation de la requête

Dans les deux précédant points nous avons défini les deux principales parties de la requête que nous allons finaliser en les regroupent de la façon suivante :

SELECT « Partie 1 » ON COLUMNS, « Partie 2 » ON ROWS FROM « Nom du Cube »

```
def strMDX(p1, p2):
    if p2 == '':
        mdx = 'SELECT {0} ON COLUMNS FROM [{1}]'.format(p1, cube)
    else:
        mdx = 'SELECT {0} ON COLUMNS, {1} ON ROWS FROM [{2}]'.format(p1, p2, cube)
    return mdx
```

Figure 31: lignes de code de la finalisation de la requête

3.3. Exécution de la requête

L'exécution de la requête suit les étapes suivantes :

- **Connexion avec ADOMD.NET**

Une fois la requête finalisée, son exécution sur la base de données et la récupération des résultats se fera par le biais d'une connexion Adomd. Nous avons intégré ces fonctionnalités à l'aide de Python.NET.

Python.NET (pythonnet) est un package qui offre aux programmeurs Python une intégration presque transparente avec le Common Language Runtime (CLR) .NET. Python.NET fournit un puissant outil de création de scripts d'application pour les développeurs .NET. Il permet de créer un script d'applications .NET ou de créer des applications entières en Python.

- **Conversion des résultats**

Les résultats de l'exécution de la requête MDX émis par le client adomd sont récupérés dans un data set que nous avons traité et transformé en un data frame puis en tableau HTML pour faciliter sont affichage à l'utilisateur final.

3.4. Interface web

Dans ce qui suit nous allons présenter le front-end de notre système (interface utilisateur).

- **Page principale**

Nous avons opté pour une interface simple et facile à utiliser. Figure 30

Figure 32 Page principale

- **Page d'erreur (**Hors contexte / Table vide**)**

Dans le cas où la requête introduite par l'utilisateur et hors contexte, cette page Figure 31 lui sera affichée et lui permettra de revenir à la page principale pour réintroduire une requête. Dans le cas où la requête exécutée retourne une table vide cette même page est affichée en précisant que la table retournée est vide.

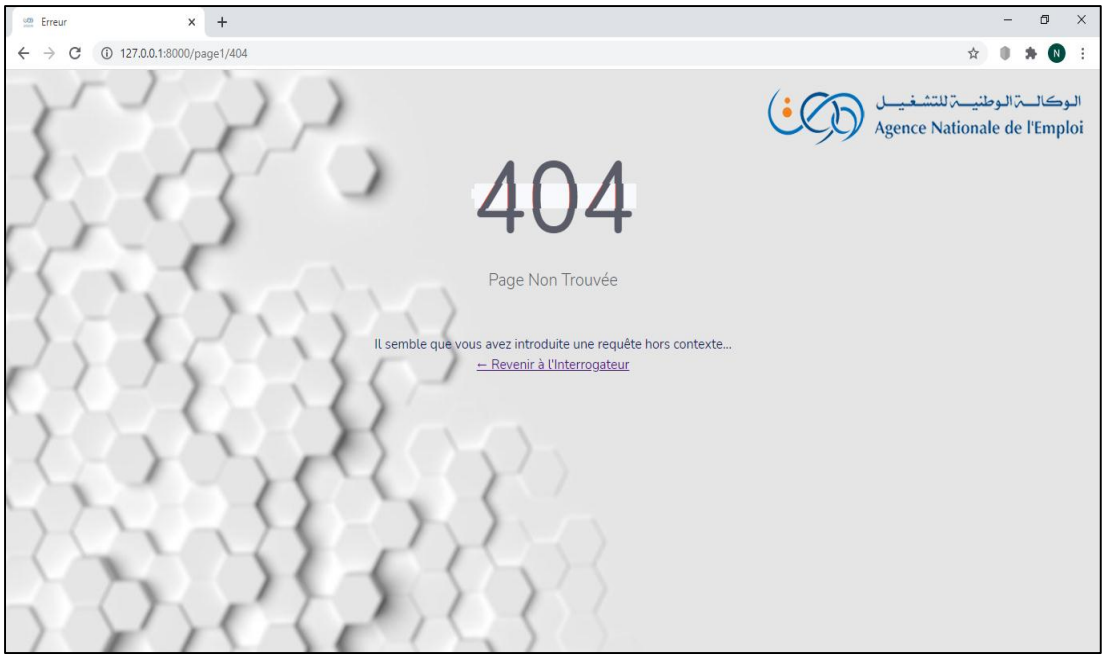
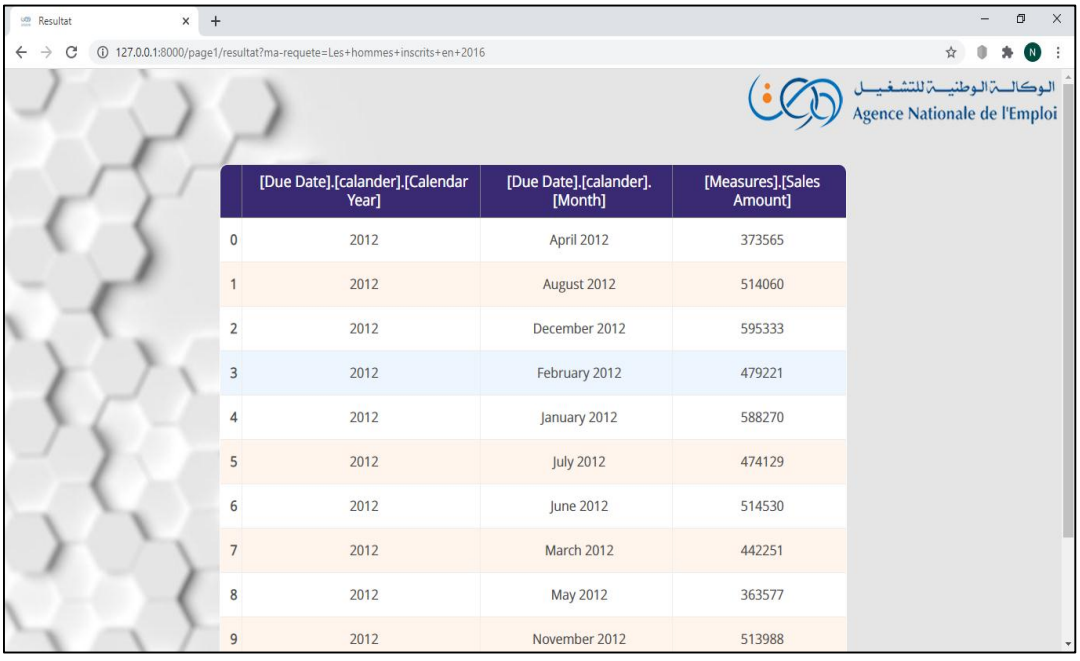


Figure 33 Page d'erreur

- **Page de résultat**

La Figure 32 représente la page de résultat qui comporte un tableau contenant les résultats de la requête introduite par l'utilisateur.



	[Due Date].[calander].[Calendar Year]	[Due Date].[calander].[Month]	[Measures].[Sales Amount]
0	2012	April 2012	373565
1	2012	August 2012	514060
2	2012	December 2012	595333
3	2012	February 2012	479221
4	2012	January 2012	588270
5	2012	July 2012	474129
6	2012	June 2012	514530
7	2012	March 2012	442251
8	2012	May 2012	363577
9	2012	November 2012	513988

Figure 34 Page de résultats

4.Conclusion

Dans ce dernier chapitre nous avons présenté les différentes étapes de réalisation de notre système qui représente un interrogateur de base de données dans le système décisionnelle de l'ANEM en langage naturel.

Nous avons expliqué les étapes d'implémentation de ce système en commençant par les outils utilisés, passant par la mise en œuvre où nous avons expliqué la création du chatbot, la traduction en requête MDX, l'exécution et terminant avec l'interface web utilisateur.

Conclusion générale

1. Conclusion générale :

Face à la mondialisation de l'informatique décisionnelle et à la concurrence grandissante, la prise de décision est devenue cruciale pour les dirigeants d'entreprises. L'efficacité de cette prise de décision repose sur la mise à disposition d'informations pertinentes.

À son terme, ce mémoire nous a permis d'exposer le déroulement de notre projet qui consiste à la " Interrogation du Système d'Information Décisionnel de l'ANEM en Langage Naturel " dans le but de faciliter l'accès à l'information pertinente,

A travers notre projet, nous avons pu réaliser un système Q&A capable de générer automatiquement des requêtes MDX adéquates aux questions posées par les utilisateurs en langage naturel.

Dans notre système le module d'analyse est réalisé par un le chatbot dialogflow ; ce dernier est sensé d'extraire des mots clés de la question posé par l'utilisateur, ces paramètres sont utilisés pour la génération automatique d'une requête MDX adéquate. La recherche d'information dans la base de données est guidée par la requête MDX généré, cela se fait par chercher les mesures et les dimensions cité dans la clause « SELECT » sur le cube désigné par la clause « FROM » afin d'extraire les informations désires et les retourner sous forme de tableau.

Notre système est de domaine fermé, l'interrogation se fait sur des données de secteur de l'emploi. Le système répond à des questions ficoïdes qui ne nécessitent qu'une seule réponse. Ce système est monolingue ; la requête de l'utilisateur, la BD et la réponse du système sont exprimés en langue française.

Ce mémoire est organisé en deux grandes parties. La première partie est consacrée à la partie théorique, dans le premier chapitre nous avons abordé les systèmes questions réponses et la notion des chatbots. Dans le deuxième chapitre nous définissons les systèmes d'information

décisionnelle passant par la modélisation multidimensionnelle, OLAP et le langage de requête MDX.

Dans la deuxième partie, nous avons proposé une architecture répondant aux problématiques posées. Le troisième chapitre et consacrer à la conception et la modélisation de notre solution proposée et le quatrième chapitre à la réalisation et l'implémentation de cette solution tout en spécifiant l'environnement de travail.

2. Perspectives :

Le prototype que nous avons mis en place n'est pas celui qui contient parfaitement toutes les fonctionnalités qu'un interrogateur de BD en langage naturel devrait avoir. Il pourrait être amélioré selon plusieurs points de vue :

- Intégrer la totalité des mesures et dimensions.
- Améliorer la conversion de la phrase en LN en requête MDX de façon à générer des requêtes MDX plus complexe.
- Mettre en place une base de connaissance pour mapper chaque phrase en LN avec son équivalent en requête MDX.

Ce projet fait partie d'un effort continu de recherche impliquant différents groupes pour espérer fournir, dans un long terme, une plateforme intégrée permettant à l'ANEM d'avoir une nette visibilité sur les fluctuations du marché de l'emploi et d'exploiter toutes les informations de la BD en l'interrogeant d'une façon simple en LN. Dans un autre registre, ce stage nous a permis d'élargir et approfondir nos connaissances sur les bases de données multidimensionnels, le langage de requête MDX, etc. IL nous a aussi rendus plus ambitieuses et plus motivées pour continuer dans ce travail de recherche.

Références bibliographiques

References bibliographies

[1] B. R. ADJANI Nassiba, “Etude de mise en place d’un plan de reprise d’activité,” Université d'Alger 1 Benyoucef BENKHEDDA Alger, Algérie, 2018.

[2] J. FRANKENFIELD, “Artificial Intelligence (AI),” 13 3 2020. [Online]. Available: https://www.investopedia.com. [Accessed 2 09 2020].

[3] Y. Boubekeur , “Identification automatique de mots clés dans les,” Université de Djilali BOUNAÂMA, Khemis Miliana, 2016.

[4] N. Kuchmann-Beauger, “Question Answering System in a Business Intelligence,” HAL, Paris, 2013.

[5] B. Ojokoh and E. Adebisi, “A Review of Question Answering Systems,” 2019.

[6] R. Satapathy, “Question Answering in Natural Language Processing [Part-I],” 2018 08 2018. [Online]. Available: https://medium.com. [Accessed 11 09 2020].

[7] “Qu'est-ce qu'un chatbot: définition et guide,” 14 09 2020. [Online]. Available: https://sendpulse.com. [Accessed 3 10 2020].

[8] “Le Chatbot expliqué à ma grand-mère,” [Online]. Available: https://www.marketing-management.io. [Accessed 3 10 2020].

[9] E. Adamopoulou and L. Moussiades, “An Overview of Chatbot Technology,” Agios Loukas, 2020.

[10] B. Ahlem and E. Narimene, “Mise en place d’une plateforme d’aide à la décision pour l’activité du département Marketing au niveau d’Algérie Télécom,” alger, 2018.

[11] “BI - Business Intelligence,” [Online]. Available: https://www-igm.univ-mlv.fr. [Accessed 19 08 2020].

[12] “L’informatique décisionnelle — B.I.,” [Online]. Available: https://perso.univ-lyon1.fr. [Accessed 12 09 2020].

[13] N. BENAOUDA and M. AMICHE, “Développement d'une solution SOLAP pour la gestion des événements du réseau routier dans la wilaya de Mostaganem,” Mostaganem.

[14] S. Lilia, “chp3-modlisation-multidimensionnelle,” Slideshare, 27 févr 2014. [Online]. Available: https://fr.slideshare.net/LiliaSfaxi. [Accessed 11 2020].

[15] “OLAP and query language: how to write OLAP queries,” [Online]. Available: https://galaktika-soft.com. [Accessed 15 11 2020].

[16] “Entropot de Donnees,” [Online]. Available: http://www-igm.univ-mlv.fr. [Accessed 10 10 2020].

[17] “SGBD et Datawarehouse,” [Online]. Available: http://www-igm.univ-mlv.fr. [Accessed 01 01 2021].

[18] "multidimensional expressions (MDX)," 4 2012. [Online]. Available: https://searchsqlserver.techtarget.com. [Accessed 02 1 2021].

[19] B. ESPINASSE, "Entrepôts de données : Introduction au langage MDX," Ecole Polytechnique Universitaire de Marseille, 2015.

[20] "djngo," [Online]. Available: https://www.djangoproject.com. [Accessed 3 12 2020].

[21] "Dialogflow," [Online]. Available: https://cloud.google.com. [Accessed 20 12 2020].

[22] "Système de Gestion de Base de Données," [Online]. Available: https://sql.sh/sgbd. [Accessed 20 12 2020].

[23] "What is SQL Server Management Studio (SSMS)?," 09 11 2019. [Online]. Available: https://docs.microsoft.com. [Accessed 20 12 2020].

[24] "ADOMD.NET," 29 03 2019. [Online]. Available: https://docs.microsoft.com. [Accessed 29 12 2020].

[25] B. L, "Python : tout savoir sur le principal langage Big Data et Machine Learning," 11 12 2020. [Online]. Available: https://www.lebigdata.fr. [Accessed 20 12 2020].

[26] JDN, "HTML (HyperText Markup Langage) : définition, traduction," [Online]. Available: https://www.journaldunet.fr. [Accessed 29 12 2020].

[27] “CSS,” [Online]. Available: http://glossaire.infowebmaster.fr. [Accessed 29 12 2020].

[28] “Définition Datawarehouse,” [Online]. Available: https://actualiteinformatique.fr. [Accessed 3 12 2020].