République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université A. Mira de Béjaïa

Faculté des Sciences Exactes

Département d'Informatique

Mémoire de Fin d'Etude

En vue de l'obtention d'un Master en Génie Logiciel

Thème

Intégration des données dans un contexte Big Data

Réalisé par M^{elles} :

AIT ALI YAHIA Yasmine

HARROU Amel

Soutenu le 07 Septembre devant le jury composé de :

Présidente :	M^{me} *BOUKERRAM* Samira	M.A.A Université de Béjaïa.
Examinatrice :	M^{me} *GHANEM* Souhila	M.C.B Université de Béjaïa.
Encadrante :	M^{me} *EL BOUHISSI* Houda	M.C.A Université de Béjaïa.

Promotion 2019 - 2020.

Remerciements

Nous vous remercions M^{me} EL BOUHISSI Houda de nous avoir consacré de votre temps. Vos recommandations nous ont permises la réalisation de ce modeste travail. Nous tenons à remercier également chacun des membres du jury pour l'intérêt porté à ce travail et d'avoir accepté de l'évaluer.

Dédicaces

Nous dédions ce modeste travail à nos familles et à nos amis qui nous ont portés et soutenus.

Nous sommes redevables à nos chers parents pour leur soutien moral et matériel et aussi, pour leur amour.

Table des matières

Table des figures

Liste des tableaux

Liste des abréviations

ATM	**A**ir **T**raffic **M**anagement.
BQL	**B**ridge **Q**uery **L**anguage.
CSV	**C**omma **S**eparated **V**alues.
COVID 19	**CO**rona **VI**rus **D**isease 2019.
DB	**D**ata **B**ase.
DC	**D**ublin **C**ore.
DL	**D**escription **L**ogic.
ETL	**E**xtract **T**ransform **L**oad.
GILS	**G**overnment **I**nformation **L**ocator **S**ervice.
GPS	**G**lobal **P**ositioning **S**ystem.
HTML	**H**yper **T**ext **M**arkup **L**anguage.
IDC	**I**international **D**ata **C**orporation.
IDO	**I**nternet **D**es **O**bjets.
JSON	**J**ava**S**cript **O**bject **N**otation.
MARC	**MA**chine **R**eadable **C**ataloging.
M2Onto	**M**ongoDB **T**o **O**ntology.
NoSQL	**N**ot **o**nly **S**tructured **Q**uery **L**anguage.
NUMPY	**NU**M**e**rical **PY**yton.
OWL	**O**ntology **W**eb **L**anguage.
RDF	**R**esource **D**escription **F**ramework.
RDQL	**R**DF **D**ata **Q**uery **L**anguage.
REDIS	**RE**mote **DI**ctionary **S**erver.
SERQL	**S**tructured **E**ntity **R**elationship **Q**uery **L**anguage.
SGBDR	**S**ystème de **G**estion de **B**ases de **D**onnées **R**elationnelles.
SPARQL	**SPA**RQL **P**rotocol and **R**DF **Q**uery **L**anguage.
SQL	**S**tructured **Q**uery **L**anguage.
TSV	**T**abulation **S**eparated **V**alues.
W3C	**W**orld **W**ide **W**eb **C**onsortium.
XML	e**X**tensible **M**arkup **L**anguage.

Résumé

Ces dernières années, la quantité de données générées par les machines physiques connectées à Internet a augmenté exponentiellement, puisque nous numérisons même les informations les plus insignifiantes, nous appelons ces données les Big Data, c'est-à-dire les données massives, ces dernières étant non seulement très volumineuses mais aussi très diverses. Ceci crée d'ailleurs, un problème de stockage, d'analyse, de traitement et surtout d'intégration des données, ce qui constitue un défi complexe pour les organisations qui déploient de grandes architectures de données en raison de la nature hétérogène des données qu'elles utilisent, par conséquent, une approche globale est primordiale pour négocier les défis de l'intégration. En effet, les ontologies sont largement utilisées dans l'intégration des données car elles représentent la connaissance comme une description formelle d'un domaine d'intérêt. Dans le domaine de la santé, les Big Data sont l'ensemble des données sanitaires et sociodémographiques disponibles auprès de différentes sources et collectées pour diverses raisons. L'utilisation de ces données hétérogènes présente de nombreux intérêts : identification des facteurs de risque d'une maladie, aide au diagnostic, choix et suivi de l'efficacité des traitements, épidémiologie, etc. ... De nombreuses technologies et outils ont été développés pour permettre l'intégration des données dans le secteur sanitaire. Notre présentation passe en revue les principales approches liées à l'intégration des données, propose une nouvelle approche qui exploite la sémantique pour résoudre le problème de la variété des Big Data. Nous décrivons particulièrement une approche permettant d'intégrer des données provenant de plusieurs types de sources afin d'améliorer la prévision de la santé.

Mots clés : Données massives, santé, hétérogénéité, intégration, ontologie.

Abstract

In recent years, the amount of data generated by physical machines connected to the Internet has increased exponentially, since we digitize even the most insignificant information, we call this data the Big Data, which are not only very large but also very diverse. This creates a problem of data storage, analysis, processing and especially data integration, which is a complex challenge for organizations that deploy large data architectures due to the heterogeneous nature of the data they use, therefore, a global approach is paramount to negotiate the challenges of integration. Indeed, ontologies are widely used in data integration because they represent knowledge as a formal description of a domain of interest. In the field of healthcare, Big Data is the set of health and socio-demographic data available from different sources and collected for various reasons. The use of these heterogeneous data is of many interests : identification of risk factors for a disease, aid in diagnosis, choice and monitoring of the effectiveness of treatments, epidemiology, etc ... Numerous technologies and tools have been developed to enable the integration of data in the health sector. Our presentation reviews the main approaches related to data integration, proposes a new approach that exploits semantics to solve the problem of Big Data variety. In particular, we describe an approach to integrate data from several types of sources to improve health forecasting.

Keywords : Big Data, health, heterogeneity, integration, ontology.

Chapitre 1

Introduction et problématique

Sommaire

1.1 Introduction :

Les Big Data sont un phénomène qui a vu le jour avec l'émergence de données volumineuses que nous ne pouvions pas traiter avec des techniques traditionnelles. Les premiers projets des Big Data sont ceux des acteurs de la recherche d'information sur le web (moteurs de recherche) tels que Google et Yahoo. En effet, ces acteurs étaient confrontés aux problèmes de la scalabilité des systèmes et du temps de réponse aux requêtes utilisateurs [1].

Très rapidement, d'autres sociétés ont suivi le même chemin comme Amazon et Facebook. Les Big Data sont devenues une tendance incontournable pour beaucoup d'acteurs industriels par rapport à ce qu'elles offrent comme qualité de stockage, de traitement et d'analyse de données.

Avec l'explosion des quantités de données numériques dans divers domaines de ces données massives, le besoin de développer une sémantique des données pour mieux les gérer, augmente exponentiellement. Les ontologies sont importantes dans la recherche informatique, en particulier dans la sémantique des données. Elles sont utilisées pour représenter une variété de données sur un domaine comme un ensemble de concepts, en utilisant un vocabulaire partagé pour indiquer les types et les propriétés de ces concepts et leurs relations. Les bases de données NoSQL (Not Only Structured Query Language) sont utilisés pour le stockage des Big Data. Elles permettent de stocker de grands volumes de données structurées, semi-structurées et non structurées. Ces bases de données souffrent d'un manque de sémantique car elles sont capables de traiter des données non structurées. Pour résoudre ce problème, nous proposons de cerner l'utilité de l'ontologie dans une base de données NoSQL [2].

1.2 Contexte et problématique :

Nous envisageons le scénario suivant, l'un des secteurs les plus courants dans le cas de données volumineuses est le secteur sanitaire. Un patient peut se rendre dans plusieurs instituts qui stockent leurs données, mais, après accord, un institut devrait pouvoir partager les données avec d'autres. Cela nécessite une énorme quantité d'intégration de systèmes, et dans ce cas, nous devrions souvent traiter avec des systèmes hérités, donc nous devons utiliser des solutions d'intégration hybrides pour recevoir des données héritées et des données provenant également de logiciels ou du cloud. La solution d'intégration doit transférer les données au bon endroit, afin que les analystes de données ou les médecins puissent y accéder.

L'intégration et l'interopérabilité des données seront au centre de leurs préoccupations car les instituts peuvent avoir des techniques de gestion des données différentes avant la fusion, et l'échange de données impliqué est énorme. L'intégration des données joue un rôle clé dans la détermination de l'efficacité de l'organisation résultante, soit au niveau de l'intégration des systèmes en arrière-plan, soit au niveau de l'intégration des processus, des tâches administratives et des bases de données. La complexité de l'intégration et de l'interopérabilité des données concerne le stockage des données, leur structure et les

moyens permettant de les intégrer et de les exploiter en tant qu'entité unique.

Examinons cet exemple : le ministère de l'enseignement supérieur en Algérie gère trente quatre (34) universités à travers le pays qui déploient des architectures Big Data et certaines structures qui fonctionnent avec des bases de données traditionnelles. Nous supposons que le ministère souhaite fusionner les activités de ces universités pour fonctionner de manière centralisée afin d'offrir de meilleurs services aux étudiants (candidature au Master, ... etc).

Dans cet exemple également, l'intégration et l'interopérabilité des données seront au centre de leurs préoccupations car les universités peuvent avoir des techniques de gestion des données différentes avant la fusion.

1.3 Objectifs et contributions :

Ce travail s'est concentré sur la création des ontologies pour l'intégration des Big Data. Notre approche est basée sur des bases de données NoSQL et des ontologies modulaires. Nous nous sommes concentrés sur les défis que les données volumineuses posent à l'intégration des données. Nous avons abordé dans l'état de l'art, l'intégration des données massives. L'intention est d'améliorer des approches pour l'intégration de données adaptée aux caractéristiques des Big Data.

Notre approche permet un accès rapide et facile aux données grâce à une vue globale utilisant le SPARQL (Sparql Protocol And Rdf Query Language), qui est un langage de requête et un protocole qui permet de rechercher, d'ajouter, de modifier ou de supprimer des données RDF (Resource Description Framework) disponibles à travers Internet, c'est l'une des technologies clés du Web sémantique présentée par le W3C (Word Wide Web Consortium).

Les ontologies ont toujours été proposées comme une solution aux problèmes d'interopérabilité et d'intégration de données, même avant l'avènement des Big Data. Le but de notre approche, est d'expliquer comment la sémantique des données peut soutenir la

gestion de ces données massives.

Notre approche est constituée de cinq (05) étapes baptisées comme suit : collection des données, nettoyage des données, analyse des données, enrichissement de l'ontologie globale, construction des requêtes.

1.4 Méthodologie de travail :

La démarche adoptée pour notre travail est guidée par de nombreuses questions issues des préoccupations de la communauté de l'intégration des données.

Etant donné que la problématique de l'intégration est complexe, nous avons proposé un cadre méthodologique relativement global et suffisamment complet pour mieux aider et guider les utilisateurs dans leur processus.

Notre démarche de travail repose plus précisément sur les étapes suivantes :

- **Etape de recherche et d'analyse :** qui établit un état de l'art des différentes technologies proposées dans le cadre de l'intégration des données et qui fait une comparaison des avantages et inconvénients de chaque approche proposée.

- **Etape d'identification du problème et de la proposition d'une solution :** qui définit la problématique et la solution proposée en vigueur.

- **Etape d'implémentation et d'expérimentation des systèmes proposés :** qui met en évidence le système proposé, son fonctionnement et son intérêt, accompagnée d'une étude de cas pour la validation.

1.5 Organisation du mémoire :

La suite du mémoire est constituée de sept (07) chapitres organisés comme suit :

Dans le deuxième chapitre, nous présenterons les concepts phares des Big Data et leur relation avec l'intégration des données qui est le centre de notre recherche. Nous y retrouverons une définition détaillée de ces données massives, ainsi que leurs origines, nous présenterons également les caractéristiques des Big Data, Par la suite, nous aborderons les bases de données NoSQL et leurs types, puis nous verrons la relation entre l'intégration et les Big Data et enfin, les challenges de ces données volumineuses.

Dans le troisième chapitre, nous aborderons des notions fondamentales d'un sujet pas de moindre importance dans notre recherche : les ontologies. Nous commencerons par diverses définitions que les chercheurs ont attribuées à ces dernières et leurs origines, par la suite, nous présenterons les différents composants de ces ontologies, ainsi que leur rôle et leurs langages de description, enfin leur cycle de vie.

Dans le quatrième chapitre, nous élaborerons l'état de l'art qui représentera tous les travaux connexes que nous synthétiserons, nous présenterons ceci dans un tableau qui contiendra les grandes lignes de chaque document synthétisé, tout en suivant chaque travail par un bref paragraphe qui le résume, par la suite, nous procéderons à une analyse comparative entre les approches des documents connexes et notre approche.

Dans le cinquième chapitre, nous présenterons en détail notre approche d'intégration de données dans un contexte Big Data utilisant les ontologies comme modèle sémantique.

Dans le sixième chapitre, nous aborderons les différents aspects liés à l'implémentation des prototypes que nous développerons, à savoir, les technologies et les logiciels choisis pour l'implémentation de notre approche.

Enfin, ce mémoire se clôturera par un septième chapitre qui proposera une synthèse et un bilan du travail effectué durant ce projet de recherche et un ensemble de perspectives

liées notamment à la poursuite de ce travail.

Chapitre 2

Big Data

Sommaire

2.1 Introduction :

Des analyses récentes ont démontré que les quantités de données générées par des machines physiques connectées à Internet croissent exponentiellement, car nous numérisons aujourd'hui jusqu'à l'information la plus insignifiante.

Actuellement nous produisons annuellement une masse de données très importante estimée à près de trois (03) trillions d'octets de données. Selon le rapport IDC (International Data Corporation), la masse totale des données créée dans le monde pour l'année 2011 était de un virgule huit (1,8) zettaoctets, et s'accroît d'un facteur de neuf (09) tous les cinq ans. Cet accroissement des données touche tous les secteurs, tant scientifiques qu'économiques, ainsi que le développement des applications Web et les réseaux sociaux [3].

Dans ce chapitre, nous présenterons des notions fondamentales sur les Big Data, à savoir, leurs origines ainsi que leurs caractéristiques. Par la suite, nous aborderons les bases de données NoSQL et leurs types, puis nous verrons la relation entre l'intégration et les Big Data et enfin, les challenges de ces données volumineuses.

2.2 Origines :

Le développement d'Internet et la multiplication des objets connectés à travers le monde s'accompagnent d'une croissance exponentielle de données. Les informations disponibles sur Internet ne sont plus seulement volumineuses, elles sont également très diverses [4]. C'est ce qui a obligé les chercheurs à trouver de nouvelles manières de voir et d'analyser le monde. Il s'agit de découvrir de nouveaux ordres de grandeur concernant la capture, la recherche, le partage, le stockage, l'analyse et la présentation des données. Ainsi sont nées les «Big Data». Il s'agit d'un concept permettant de stocker un nombre indicible d'informations sur une base numérique. Cette appellation est apparue en octobre 1997.

L'arrivée des Big Data est maintenant présentée par de nombreux articles comme une nouvelle révolution industrielle semblable à la découverte de l'électricité et de l'informatique [5].

2.3 Définition :

Big Data signifient méga données, grosses données ou encore données massives. Ils désignent un ensemble très volumineux de données qu'aucun outil classique de gestion de

base de données ne peut vraiment traiter. En effet, nous procréons environ deux virgule cinq (2,5) trillions d'octets de données tous les jours. Ce sont les informations provenant de divers outils numériques : messages que nous envoyons, vidéos que nous publions, informations climatiques, signaux GPS (Global Positioning System), enregistrements transactionnels d'achats en ligne et bien d'autres encore. Les géants du Web, au premier rang : Yahoo, Facebook et Google, ont été les tous premiers à déployer ce type de technologie pour la facilité d'accès à leurs bases de données géantes [6].

2.4 Caractéristiques :

Le concept «Big Data» regroupe une famille d'outils qui répondent à une triple problématique dite règle des 3Vs : le Volume, la Vitesse et la Variété, auxquels s'ajoutent d'autres "V" complémentaires, comme ceux de Valeur et de Validité [6] :

- **Volume :** les Big Data sont associées à un volume de données vertigineux, se situant actuellement entre quelques dizaines de téraoctets (1 To = 212 octets) et plusieurs pétaoctets (1 Po = 215 octets) en un seul jeu de données. Le volume correspond à la masse d'informations produite chaque seconde. Les entreprises gérant des données massives, se voient assujetties à trouver des techniques nécessaires pour gérer les volumes de données collectés chaque jour.

- **Vitesse (Vélocité) :** elle décrit la fréquence à laquelle les informations sont générées, capturées, stockées et partagées. Elle est aussi le traitement des flux continus de données. Les entreprises doivent appréhender la vitesse non seulement en termes de création de données, mais aussi sur le plan de leur traitement, de leur analyse et de leur restitution à l'utilisateur en respectant les exigences des applications en temps réel.

- **Variété :** de plus en plus, le taux des données structurées manipulées dans des tables de bases de données relationnelles est en décroissance par rapport à l'expansion des types de données non-structurées. Les technologies Big Data, permettent de faire la création, l'intégration, l'analyse, la reconnaissance, le classement des don-

nées de différents types.

- **Véracité (Validité) :** l'aptitude à juger la crédibilité et la fiabilité du nombre indéfini de données collectées. Il est difficile de justifier l'authenticité et l'exactitude des contenus des différents volumes et variétés de données manipulées.

- **Valeur :** dans un contexte d'infobésité, il s'agit d'être capable de se concentrer sur les données ayant une réelle valeur et étant exploitables pour justifier leur analyse.

2.5 Types de données :

Il existe plusieurs types de données dans les Big Data :

- **Données structurées :** ce sont des données qui adhèrent à un modèle de données prédéfini et qui sont donc faciles à analyser, à accéder et à capturer comme les bases de données relationnelles et les feuilles de calcul. Pour cette raison, les données structurées apportent des avantages inhérents lors du traitement de grands volumes d'informations [7].

- **Données semi-structurées :** elles sont principalement caractérisées par le fait qu'elles aient une structure non uniforme et implicite, qui peut évoluer rapidement comme la messagerie électronique, l'XML (eXtensible Markup Language). Ces données sont exprimées dans des formats tels que XML et JSON (JavaScript Object Notation), qui permettent la représentation des informations sous forme hiérarchique (c'est-à-dire une structure arborescente) en utilisant des balises ou des symboles en tant qu'éléments séparateurs [8].

- **Données non-structurées :** toute donnée dont la forme ou la structure est inconnue est classée comme donnée non structurée comme les données émanant des réseaux sociaux, les vidéos et les images. Outre la taille énorme, les données non structurées posent de nombreux problèmes en termes de traitement [9].

2.6 Base de données NoSQL :

Il n'est pas du tout facile de traiter d'énormes volumes de données très diversifiées, complexes et rapides. Les outils traditionnels de gestion des données n'ont pas les performances requises pour traiter ces données. En revanche, les bases de données NoSQL regroupent des données hétérogènes. Ils permettent de stocker de grands volumes de données structurées, semi-structurées et non structurées. De plus, elles fournissent un accès à grande vitesse aux données semi-structurées et non structurées et sont très flexibles. Ainsi, les bases de données NoSQL résolvent en partie, les problèmes de volume, de variété et de vitesse mais ne traitent pas la sémantique [2].

Les bases de données NoSQL appelées également Cloud Data Bases, no-relational DataBases ou alors Big Data DataBases ne sont pas dotées d'un schéma et d'une interface de requêtes en SQL (Structured Query Language).

Il existe différents types de bases de données NoSQL :

2.6.1 Clés/Valeurs :

- Le but de la famille clé-valeur est la flexibilité et la simplicité.
- Les bases de données clés-valeurs consistent à associer des clés à des valeurs.
- La clé identifie la donnée de manière unique et permet l'accès et la gestion de la donnée.
- La valeur stockée peut avoir plusieurs types : entier, chaine de caractères, JSON, XML, HTML (Hyper Text Markup Language), images, vidéos ...etc [10][11].

Exemples : Amazon BynamoDB (DataBase), Azure Table Storage, Riak, REDIS (REmote DIctionary Server).

2.6.2 Orientées colonnes :

- Évolution de la base de données clé/valeur.
- Ressemblent aux systèmes de gestion de bases de données relationnelles.

- Les données sont organisées en colonnes et aucune colonne ne contient la valeur NULL.
- Le nombre de colonnes est dynamique [10].

Exemples : Apache HBase, Cassandra.

2.6.3 Orientées documents :

- Étendent le paradigme clé/valeur, avec des documents plus complexes à la place des données simples, et une clé unique pour chacun d'eux.
- Les documents sont de type JSON ou XML.
- L'accès aux données peut se faire à l'aide d'une seule clé [10].

Exemples : MongoDB, Couchbase Server, CouchDB, RavenDB.

2.6.4 Orientées graphes :

- Basées sur les théories des graphes.
- Adaptées aux traitements des données des réseaux sociaux.
- Ces bases de données sont utilisées lorsque les données peuvent être représentées sous forme de graphe.
- Le graphe se compose de nœuds et de bords, où les nœuds agissent en tant qu'objets et les bords agissent en tant que relation entre les objets [10][11].

Exemples : Neo4J, InfiniteGraph, OrientDB.

2.7 Intégration des Big Data :

L'intégration des données est un ensemble de processus utilisé pour récupérer et combiner des données provenant de sources disparates en informations significatives et précieuses. Une solution complète d'intégration des données permet d'obtenir des données fiables provenant de diverses sources. Les techniques traditionnelles d'intégration de données étaient principalement basées sur le processus ETL (Extract Transform Load), pour ingérer et nettoyer les données puis les charger dans un entrepôt de données. Aujourd'hui, un énorme volume de données est collecté à partir de nombreuses sources de données

hétérogènes qui génèrent des données en temps réel avec différentes qualités, ce que nous appelons les Big Data. L'intégration des données volumineuses est très difficile, d'autant plus que les techniques traditionnelles d'intégration des données n'ont pas réussi à la gérer.

L'objectif de l'intégration des données est de fournir un accès unifié aux données qui nécessitent des informations provenant de sources multiples et de fournir aux utilisateurs une vue unifiée des données.

Ziegler et Dittrich dans [12] expliquent les raisons qui sous-tendent l'intégration de sources de données multiples, qui sont :

- Faciliter l'accès à l'information en fournissant une vue intégrée d'un ensemble de systèmes d'information existants.
- Obtenir une base plus complète en combinant des données provenant de différents systèmes d'information complémentaires.

Le processus traditionnel d'intégration des données est supposé être un processus en trois (03) étapes dont la dernière est appelée fusion de données. Au cours de cette étape, les représentations doubles des données sont combinées et fusionnées en une seule image, tandis que les incohérences sont résolues. Les deux autres étapes sont le mappage du schéma et la détection des doublons [13].

L'interopérabilité des données devient une nécessité pour les applications qui doivent se présenter dans un environnement Big Data. Ce défi est accentué par l'hétérogénéité des données qu'elles utilisent. L'interopérabilité dans le contexte des données massives permet de partager des informations entre les individus, les fournisseurs et les organisations de sorte que les systèmes et les applications puissent échanger et utiliser les informations des données volumineuses sans effort particulier. Cela semble facile, mais difficile à mettre en œuvre correctement, car l'interopérabilité nécessite des bases étroitement contrôlées. En effet, les Big Data entraînent la prolifération de données provenant de nombreuses sources.

La collecte de ces données accentue la croissance des données, mais il est plus difficile de connecter ces données pour y accéder et les manipuler.

2.8 Défis des Big Data :

Les défis de l'intégration et de l'interopérabilité des grandes données sont nombreux. Certains sont énumérés ci-dessous [14] :

- **Incohérence des données :** les données provenant de sources hétérogènes pourraient entraîner une incohérence des niveaux de données, ce qui nécessite plus de ressources pour optimiser les données non structurées. Les données structurées permettent d'effectuer les opérations de requête pour analyser, filtrer et utiliser ces données pour les décisions commerciales et les capacités d'organisation. Dans ce scénario, lorsque les grands ensembles de données sont concernés, les données non structurées résident dans des volumes plus importants. Il est possible de les localiser à l'aide des méthodes d'étiquetage et de tri, ce qui permet de rechercher les données à l'aide des mots clés. De plus, les méthodologies Hadoop comme MapReduce, Yarn qui modulent les grands ensembles de données en sous-divisions pour faciliter les conversions des données et programmer les processus individuellement. Des canaux peuvent être mis en œuvre pour la diffusion en continu de grands ensembles de données.

- **Optimisation des requêtes :** optimisation des requêtes à chaque niveau de l'intégration des données et mise en correspondance des composants avec le schéma existant ou un nouveau schéma qui pourrait influencer les attributs existants et nouveaux. Ce défi peut être relevé en réduisant le nombre de requêtes par l'utilisation de chaînes de caractères, de jointures, d'agrégation, de regroupement de toutes les données relationnelles. Le traitement parallèle, où les opérations de requête asynchrones sont effectuées sur des threads individuels, peut influencer positivement la latence et le temps de réponse.

- **Ressources insuffisantes :** ressources insuffisantes pour la mise en œuvre de l'intégration des données, c'est-à-dire le manque de ressources financières, le manque de professionnels qualifiés et les coûts de mise en œuvre. Chaque organisation doit analyser ses capacités d'investissement afin de mettre en œuvre une nouvelle phase

dans son environnement de travail existant. Le manque de ressources financières est généralement les petites organisations, qui sont limitées à un domaine particulier. Exemple : une organisation limitée à la consultation. En temps réel, ces organisations peuvent mettre en œuvre les changements à intervalles rapprochés, ce qui leur permet de disposer de plus de temps pour récupérer les ressources investies. Le manque de professionnels qualifiés peut non seulement ralentir les projets, mais aussi démoraliser les capacités de l'organisation à gérer les projets. Il est difficile de trouver des professionnels compétents pour les données volumineuses, car l'intégration des données nécessite un niveau élevé de professionnels expérimentés qui, par le passé, auraient traité le module d'intégration. Cette situation peut être freinée si l'organisation met en place des modules de formation pour ses employés. Les coûts de mise en œuvre pourraient être plus élevés pour la mise en œuvre de l'intégration des données.

- **Système de soutien à la mise en œuvre :** les organisations doivent mettre en place un système de soutien pour le traitement des mises à jour et des rapports d'erreurs à chaque étape de l'intégration des données, ce qui nécessiterait un module de formation pour former les professionnels au traitement des rapports d'erreurs. Cela peut nécessiter un investissement énorme pour une organisation. Pour relever ce défi, les organisations doivent mettre en œuvre des avancées dans leur système de travail afin de s'adapter aux tendances croissantes du marché. La mise en œuvre d'un système de support pourrait aider à analyser les défauts de l'architecture existante, ce qui pourrait leur donner une marge de manœuvre pour de nouvelles mises à jour ou modifications. Bien qu'il s'agisse d'un investissement élevé, cela pourrait quand même s'avérer bénéfique pour les organisations après les changements.

L'intégration des données est également cruciale pour les collaborations entre entreprises lorsque deux entreprises veulent fusionner en une seule entité, mais que chacune utilise une source de données différente. En outre, certains projets scientifiques peuvent nécessiter des données provenant de différentes disciplines, ce qui signifie des sources de données différentes. Dans ce contexte, l'intégration des données peut aider à partager les résultats de la recherche sans la nécessité de charger les données dans une seule source de données.

2.9 Conclusion :

Dans ce chapitre, nous avons présenté l'ensemble des notions fondamentales des Big Data, notamment l'origine, les caractéristiques, les bases de données NoSQL, sans oublier les défis et l'intégration des données massives qui sont les éléments phares de notre recherche.

Dans le chapitre suivant, nous présenterons d'une manière détaillée les notions phares des ontologies qui sont le cœur de notre projet de recherche.

Chapitre 3

Ontologies

Sommaire

3.1 Introduction :

Les ontologies ont été développées par la communauté de l'intelligence artificielle pour faciliter le partage et la réutilisation des connaissances, transportant la sémantique pour des domaines particuliers. Une utilisation courante des ontologies est la normalisation et la conceptualisation des données via un langage formel d'ontologie compréhensible par les machines.

Dans ce chapitre, nous aborderons des concepts des ontologies. Nous commencerons par diverses définitions que les chercheurs ont données à ces dernières et leurs origines, par la suite, nous présenterons les différents composants de ces ontologies, ainsi que leur rôle et leurs langages de description, enfin, leur cycle de vie.

3.2 Origines :

L'ontologie est un terme qui est apparu dans la métaphysique avec Aristote qui considérait que l'ontologie est une "Science qui étudie l'être en tant qu'être et les attributs qui lui appartiennent essentiellement". Dans ce contexte, élaborer une ontologie, revient à faire l'étude philosophique de la nature, de l'être et de l'existence, c'est-à-dire, l'étude des propriétés générales de ce qui existe, en définissant l'ensemble des connaissances sur le monde [15].

3.3 Définition :

Pendant la dernière décennie, les informaticiens ont repris le terme "Ontologie" qui est devenu très utilisé dans le domaine de l'informatique. Beaucoup de définitions ont été attribuées à ce terme par plusieurs chercheurs.

Une des premières définitions a été donnée par Neches et al. (1991) : "Une ontologie définit les termes et les relations de base comportant le vocabulaire d'un domaine, aussi bien que les règles pour combiner ces termes et ces relations afin de définir des extensions du vocabulaire" [16].

La définition donnée par Studer et al. (1998) est comme suit : "Une ontologie est une spécification formelle et explicite d'une conceptualisation partagée" [17].

- Le terme "conceptualisation" désigne un modèle abstrait de certains phénomènes réels par l'identification des concepts importants caractérisant un domaine.
- Le terme "formelle" signifie qu'une ontologie doit être interprétable et lisible par la machine.
- Le terme "explicite" veut dire que les entités et les axiomes doivent être explicitement définis.
- Le terme "partagée" indique qu'une ontologie doit annoter multiples sources de données, être consensuelle et accessible par tous les utilisateurs d'une communauté particulière.

La définition donnée par Gruber est : "Une ontologie est une spécification d'une conceptualisation partagée d'un domaine". Elle peut présenter les connaissances de manière à ce qu'elles puissent être partagées et utilisées répétitivement et fournit un moyen efficace de réduire le volume de données en codant la structure d'un domaine particulier [2].

3.4 Composantes :

Une ontologie est composée des éléments suivants [18] :

- **Concepts :** aussi appelés classes, ils sont utilisés dans un sens large. Ils peuvent être abstraits ou concrets, élémentaires ou composites, réels ou fictifs. En bref, un concept peut être tout ce dont nous parlons et par conséquent, peut également être la description d'une tâche, d'une fonction, d'une action, d'une stratégie, d'un processus de raisonnement, ... etc.

- **Relations :** elles représentent un type d'interaction entre les concepts du domaine. Elles sont formellement définies comme tout sous-ensemble d'un produit de N ensembles. Nous considérons d'abord le lien entre les relations et les autres composantes de l'ontologie. Nous nous demanderons si les concepts et les attributs sont considérés, respectivement, comme des relations unitaires et binaires.

- **Fonctions :** elles sont considérées comme un type spécial de relation où la valeur du dernier argument est unique pour une liste de valeurs des N-1 arguments précédents.

- **Axiomes :** ce sont des phrases modèles qui sont toujours vraies. Elles sont incluses dans une ontologie pour plusieurs raisons, comme pour contraindre son information, pour vérifier son exactitude ou pour en déduire de nouvelles informations.

- **Instances/Faits :** les instances représentent les éléments d'un concept donné. Les faits représentent une relation qui existe entre des éléments.

3.5 Langages de description :

Une ontologie peut être représentée par plusieurs langages, dans ce qui suit, nous citons les plus fréquents [19] :

- **RDF :** acronyme de Resource Description Framework, développé et présenté par le W3C au début de 1999 pour l'intégration des métadonnées et montre également comment décrire les ressources disponibles sur Internet telles que le contenu des sites web. Le RDF soutient directement l'intégration et l'analyse parmi les applications hétérogènes de l'IDO (Internet Des Objets) qui échangent des données lisibles par la machine sur le Web.

- **RDF schema :** c'est une extension sémantique du vocabulaire RDF de modélisation des données. Il illustre les descriptions connexes et établit les liens significatifs entre les ressources.

- **OWL :** acronyme de Ontology Web Language (Langage d'Ontologie Web), ce langage a été développé par le W3C. Il s'agit d'un langage de balisage sémantique utilisé pour échanger l'ontologie sur le Web. L'objectif principal de la création de l'OWL est d'étendre les vocabulaires des données RDF. La famille OWL comprend OWL Full, OWL DL (Description Logic) et OWL Lite.

La **figure 3.1** montre la structure complète de la famille OWL :

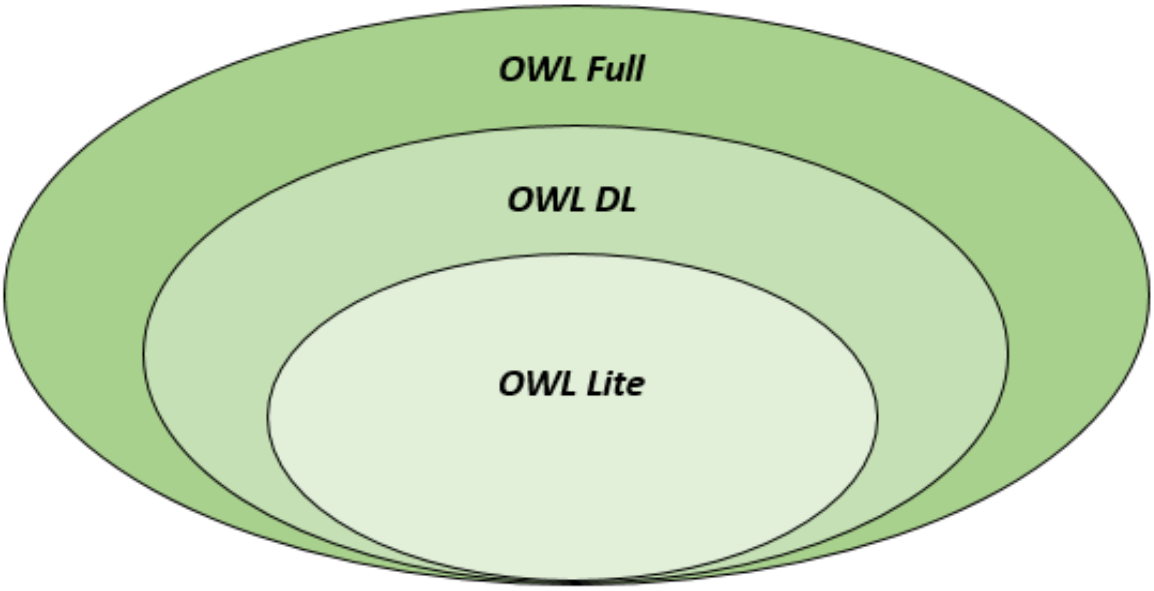


FIGURE 3.1 – Structure complète de la famille OWL.

- **OWL Full :** il utilise toutes les fonctionnalités du langage OWL. De plus, il prend les primitives de RDF et RDF Schema. Il est entièrement compatible avec RDF d'un point de vue sémantique et syntaxique.

- **OWL DL :** c'est la sous-partie, elle prend des primitives limitées par rapport à l'OWL Full. La logique de description est assurée pour soutenir le raisonnement sémantique.

- **OWL Lite :** cette fois encore, des primitives limitées ont été prises et appliquées par rapport à OWL DL. Il s'agit d'une extrémité limitée qui ne prend pas en charge les cardinalités, les classes d'énumérations et les opérations disjointes. Cependant, il est facile de comprendre la logique des primitives et des implémentations.

3.6 Cycle de vie :

Les ontologies étant destinées à être utilisées comme des composants logiciels dans des systèmes répondant à des objectifs opérationnels différents, leur développement doit

s'appuyer sur les mêmes principes que ceux appliques en génie logiciel [20].

Le cycle de vie des ontologies a été défini selon deux points de vue successifs. En effet, dans un premier temps, les ontologies ont été considérées comme des objets statiques assimilés à des composants logiciels. Dans cette phase, les recherches étaient consacrées essentiellement à la construction et à la formalisation des ontologies. Ce cycle de vie a ensuite évolué dans les années 2000. On n'a plus considéré les ontologies comme des objets statiques, mais plutôt comme des objets dynamiques qui devront être régulièrement modifiés. Dans ce sens, il ne s'agit plus de se concentrer sur leur construction, mais plutôt sur leur évolution. L'évolution des ontologies devient alors un cycle de vie à part, incluant plusieurs étapes et soulevant plusieurs problèmes [21].

Les activités liées aux ontologies sont d'une part, des activités de gestion de projet : planification, contrôle, assurance qualité, et d'autre part, des activités de développement : spécification, conceptualisation, formalisation, s'y ajoutent des activités transversales de support telles que l'évaluation, la documentation et la gestion de la configuration. Le cycle de vie d'une ontologie comprend une étape initiale d'évaluation des besoins, une étape de construction, une étape de diffusion, et une étape d'utilisation. Après chaque utilisation significative, l'ontologie et les besoins sont réévalués et cette ontologie peut être étendue, si nécessaire.

La **figure 3.2** présente le cycle de vie d'une ontologie :

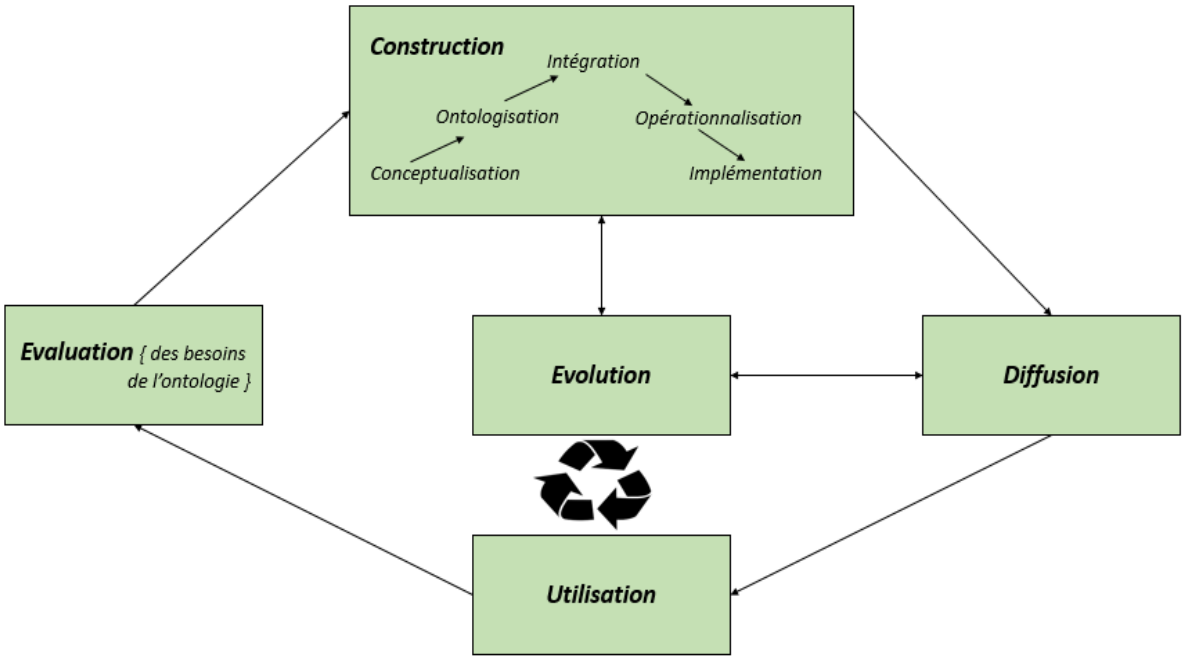


FIGURE 3.2 – Cycle de vie d'une ontologie.

La phase de construction peut être décomposée en trois (03) étapes : conceptualisation, ontologisation et opérationnalisation. L'étape de l'ontologisation peut être complétée par une étape d'intégration dans laquelle une ou plusieurs ontologi seront importées dans l'ontologie à construire [20].

La **figure 3.3** montre les étapes de la phase de construction :

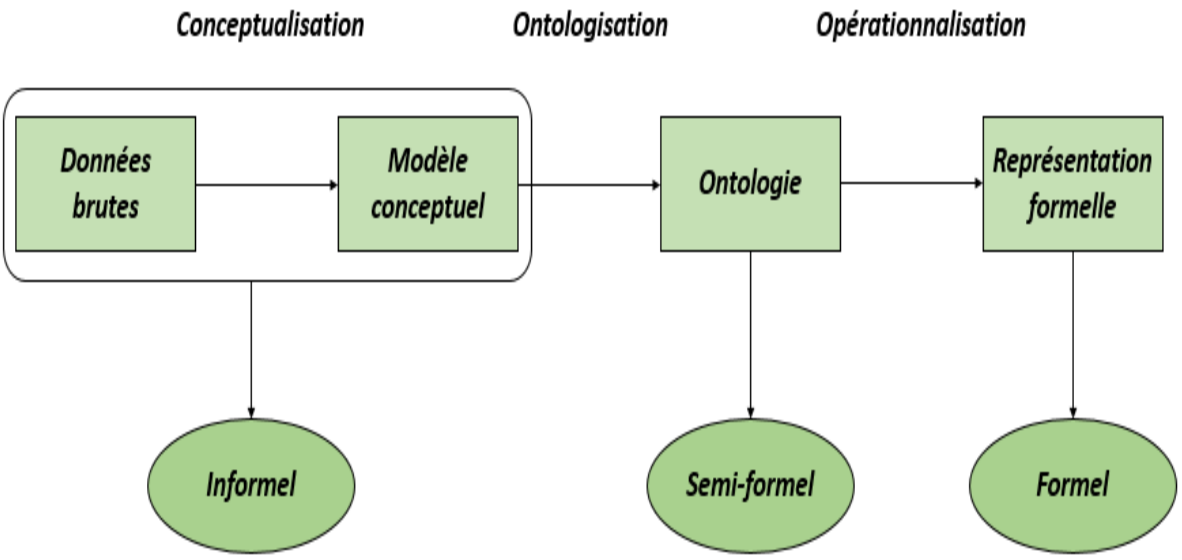


FIGURE 3.3 – Etapes de la phase de construction.

— **Evaluation des besoins :** le but de construire une ontologie se décline en trois (03) aspects :

- **L'objectif opérationnel :** il est indispensable de bien préciser l'objectif opérationnel de l'ontologie, en particulier à travers des scenarios d'usage.
- **Le domaine de connaissance :** il doit être précisément défini.
- **Les utilisateurs :** ils doivent être identifiés et en accord avec l'objectif opérationnel, le degré de formalisme de l'ontologie, et sa granularité.

Une fois le but défini, le processus de construction de l'ontologie peut démarrer, en commençant par la phase de conceptualisation.

— **Conceptualisation :** identification des connaissances contenues dans un corpus représentatif du domaine. Ce travail permet d'aboutir à une modélisation informelle voir semi-formelle, partiellement cohérente, et il doit être mené par un expert du domaine, assisté par un ingénieur de la connaissance.

— **Ontologisation :** une formalisation partielle, respectant l'intégrité du modèle conceptuel, permettra à cette étape, de construire une ontologie proprement dite. Afin de respecter les objectifs généraux des ontologies, GRUBER propose cinq (05) critères permettant de guider le processus d'ontologisation :

- **La clarté et l'objectivité des définitions :** elles doivent être indépendantes de tout choix d'implémentation.
- **La cohérence :** consistance logique des axiomes afin de pouvoir formuler par la suite, des inférences cohérentes.
- **L'extensibilité d'une ontologie :** la possibilité de l'étendre sans modification.
- **La minimalité des postulats d'encodage :**ce qui assure une bonne portabilité.
- **La minimalité du vocabulaire :** l'expressivité maximum de chaque terme.

— **Opérationnalisation :** transcription de l'ontologie dans un langage formel et opérationnel de représentation de connaissances. L'ontologie obtenue est opérationnelle dans le sens où elle peut inclure des mécanismes de raisonnement. Ce travail doit être mené par l'ingénieur de la connaissance.

3.7 Conclusion :

Dans ce chapitre, nous avons présenté l'ensemble des concepts des ontologies, notamment l'origine, les composants, les langages de description, sans oublier le rôle de ces ontologies dans notre projet de recherche et enfin, leur cycle de vie.

Dans le chapitre suivant, nous procéderons à une synthèse de plusieurs travaux connexes en élaborant un état de l'art pour les différents travaux de recherche.

Chapitre 4

Etat de l'art

Sommaire

4.1 Introduction :

Le volume, la variété et la complexité croissants des données collectées à des fins scientifiques posent des défis en matière d'intégration des données. Pour que les données soient utiles, les scientifiques doivent non seulement pouvoir y accéder, mais aussi être capables de les interpréter et de les utiliser. Pour ce faire, il faut un contexte sémantique.

L'intégration sémantique des données est le processus qui consiste à utiliser une représentation conceptuelle des données et de leurs relations pour éliminer les éventuelles hétérogénéités. En effet, il existe différentes méthodes d'intégration de données, nous verrons quelques-unes dans les travaux que nous synthétiserons plus tard dans ce chapitre.

Dans ce chapitre, nous élaborerons l'état de l'art qui représentera tous les travaux connexes que nous synthétiserons, nous présenterons ceci dans un tableau qui contiendra les grandes lignes de chaque document synthétisé, tout en suivant chaque travail par un bref paragraphe qui le résume, par la suite, nous procéderons à une analyse comparative

entre les approches des documents connexes et notre approche.

4.2 Travaux connexes :

La croissance continue du volume et de la vitesse des données avec des types de données diversifiés exige la nécessité d'utiliser les services des outils d'intégration de données pour agréger des données provenant de sources disparates.

Plusieurs approches d'intégration ont été proposées pour intégrer les grandes sources de données et pour résoudre les conflits d'interopérabilité.

Dans cette section, nous présenterons les principales approches liées à l'intégration des données dans un contexte de données volumineuses, ce qui converge avec notre travail de recherche.

Curé et al. 2013 [22], ont présenté une solution d'intégration de données, basée sur l'ontologie qui vise à intégrer les données résidant dans les systèmes NoSQL orientés documents et colonnes. Dans ce travail, un schéma local est généré pour chaque système sous forme d'ontologie, afin de définir un schéma global à partir des correspondances entre les ontologies locales. Les auteurs proposent également un BQL (Bridge Query Language) qui traduit les requêtes SPARQL vers les différents langages de requête des systèmes locaux.

Ajani, 2014 [23], a proposé une technique de recherche sémantique basée sur les métadonnées et les ontologies pour les grands domaines de recensement dans un contexte Big Data. Cette proposition s'est concentrée sur la construction d'une ontologie spécifique à ce domaine et sur la recherche sémantique pour obtenir le résultat souhaité à partir d'un ensemble volumineux de données. L'auteur affirme que la proposition peut être utile au gouvernement ainsi qu'aux organismes tiers pour améliorer la recherche sémantique sur les données de recensement et pour effectuer diverses actions et prendre des décisions. Une ontologie peut jouer un rôle important dans l'organisation de l'information distribuée liée au domaine d'intérêt spécifique. La technique proposée utilise le framework "Protégé" pour fournir une recherche sémantique basée sur l'ontologie pour un domaine spécifique.

Jirkovský et Obitko, 2014 [24], ont proposé une approche pour l'intégration des Big Data basée sur la réduction de l'hétérogénéité sémantique des données massives dans le domaine de l'automatisation industrielle. Ils traitent de l'hétérogénéité structurelle et sémantique. Pour traiter l'hétérogénéité structurelle, ils prennent en considération différents types de sources de données telles que les fichiers texte, XML et les bases de données. Pour traiter l'hétérogénéité sémantique, ils créent une ontologie partagée, qui assure la transformation des sources de données dans le même langage. La première étape de cette approche est la création semi-manuelle d'une ontologie partagée, qui assure le partage des connaissances. Les auteurs traitent d'abord de l'hétérogénéité structurelle. Ce problème figure dans l'étape de prétraitement. Les stratégies de traitement des sources de données diffèrent selon leur catégorie. L'action suivante est la construction d'une ontologie partagée à partir des données prétraitées. Une étape cruciale consiste à comprendre un contenu donné et à identifier les entités correspondantes dans toutes les sources de données. Certains systèmes de correspondance d'ontologies sont exploités pour cette tâche.

Bansal et Kagemann, 2015 [25], ont proposé un framework sémantique d'extraction, de transformation et de chargement ETL pour l'intégration des Big Data. Ce dernier proposé génère un modèle sémantique des ensembles de données en cours d'intégration, puis génère des données sémantiques liées conformément au modèle de données. L'utilisation des technologies sémantiques est introduite dans la phase de transformation d'un processus ETL pour créer un modèle de données sémantiques et générer des données liées sémantiquement (triples RDF) et les stocker dans un entrepôt de données. La phase de transformation continue à effectuer d'autres activités telles que la normalisation et le nettoyage des données, cette même phase implique un processus manuel d'analyse des ensembles de données, du schéma et de leur finalité. Sur la base des résultats, le schéma doit être mis en correspondance avec une ontologie existante spécifique au domaine où une ontologie devra être créée à partir de zéro. Si les sources de données appartiennent à des domaines disparates, plusieurs ontologies sont nécessaires et des règles d'alignement sont spécifiées pour tous les champs de données communs ou connexes. Les phases d'extraction et de chargement du processus ETL resteraient les mêmes.

Knoblock et Szekley, 2015 [26], ont introduit un nouvel outil appelé "Karma" pour l'exploitation de la sémantique pour l'intégration des Big Data dans le patrimoine culturel. La proposition est capable d'importer diverses sources de données, y compris des sources relationnelles et hiérarchiques, pour construire finalement des modèles sémantiques riches, utiles pour l'intégration. La proposition comporte quatre (04) étapes principales : la première étape consiste à importer des données provenant de diverses sources, la deuxième étape consiste à nettoyer les données, à identifier les valeurs aberrantes et à normaliser les données afin que les mêmes formats soient utilisés dans toutes les sources connexes. La troisième étape consiste à créer une description sémantique de chaque source, et le système tente d'automatiser autant que possible ce processus de modélisation des sources. Enfin, il intègre les données en convertissant tous les enregistrements dans un format uniforme à l'aide des descriptions sémantiques et en intégrant ensuite les données dans ce framework unifié. Les auteurs ont dû relever un défi majeur, à savoir importer, normaliser, modéliser et intégrer rapidement les données provenant de nombreuses sources de données muséales différentes.

Keller et al. 2016 [27], ont proposé un système permettant de combiner différentes sources de données de gestion du trafic aérien en utilisant des techniques d'intégration sémantique. Il transforme les données des formats sources originaux en une illustration sémantique standardisée dans un triplestore avec des données Sherlock basées sur l'ontologie. Dans ce contexte, SPARQL est utilisé comme un outil de recherche pour récupérer des informations sur le magasin triplestore. L'architecture principale se compose de quatre (04) étapes : la première étape permet de choisir trois origines du Sherlock pour l'intégration : les données de trajectoire de vol, les données météorologiques de l'aéroport et les informations sur les avis de circulation aérienne. La deuxième étape permet d'utiliser l'ontologie ATM (Air Traffic Management) pour ajouter l'aspect sémantique. Malheureusement, le fait de demander aux utilisateurs aéronautiques de comprendre l'ontologie ATM et d'apprendre la syntaxe SPARQL n'est pas réaliste. La troisième étape permet de convertir le format source des données ATM originales et de les traduire en triples RDF. La dernière étape permet d'interroger et de télécharger le service, mais elle n'a pas encore été mise en œuvre.

Fang et al. 2016 [28], ont proposé une approche pour les grandes données sémantiques. Elle comprend quatre couches : la couche des métadonnées, qui utilise différentes règles de métadonnées telles que MARC (Machine Readable Cataloging), DC (Dublin Core), GILS (Government Information Locator Service) pour décrire les données touristiques telles que la localisation géographique et les relations entre ces ressources. La couche d'ontologie permet d'assurer l'interopérabilité sémantique dans différents types de métadonnées. À cette fin, nous pouvons utiliser deux (02)méthodes : la première méthode permet d'intégrer les attributs et les concepts des différentes règles de métadonnées dans l'ontologie en utilisant le langage de représentation des connaissances OWL. La seconde méthode permet d'utiliser le langage ontologique pour transformer le format des métadonnées en format RDF. La couche de données liées publie les données selon le principe des données liées. Elle fournit un mécanisme d'accès unifié pour différents formats de données et produit une interopérabilité sémantique entre ces données. La couche d'application des données permet de fournir la méthode traditionnelle de recherche par mot-clé et une interface plus conviviale pour la recherche interactive.

Abbes et Gargouri 2017 [29], ont construit une ontologie OWL pour l'intégration des Big Data tout en considérant les caractéristiques de ces données massives, et ont fourni un modèle partagé pour les sources de données, en utilisant différents types de données (données structurées, semi-structurées et non structurées). Pour atteindre leurs objectifs, les auteurs ont suivi une approche basée sur trois (03) étapes principales : la première étape est d'envelopper les sources de données dans les bases de données MongoDB, la deuxième étape consiste à générer des ontologies locales, et la troisième étape revient à composer les ontologies locales pour en obtenir une globale. L'approche utilisée exploite une base de données NOSQL pour envelopper les sources de données, à savoir, MongoDB, et décrit également deux outils mettant en œuvre les deuxième et troisième étapes, soit, M2Onto (MongoDB To Ontology) pour générer des ontologies locales et MOOM pour les fusionner en une ontologie globale. Les auteurs ont dû relever un défi majeur, car les Big Data couvrent non seulement le stockage et la gestion d'une variété de données, mais aussi l'extraction de connaissances cohérentes à partir de ces données.

Vathy-Fogarassy et Hugyák, 2017 [30], ont proposé une approche de fusion de données

basée sur un méta-modèle, elle permet d'interroger simultanément des données provenant de systèmes de bases de données hétérogènes (relationnelles et NoSQL). La méthode présentée couvre les hétérogénéités structurelles, sémantiques et syntaxiques des systèmes sources en utilisant le méta-modèle. La méthode présentée a été évaluée expérimentalement en développant une application web qui permet aux utilisateurs d'interroger des données à partir des SGBDR (Systèmes de Gestion de Bases de Données Relationnelles) et de MongoDB. Ce travail propose une interface conviviale qui permet d'interroger des données provenant des systèmes de bases de données hétérogènes sans aucune compétence en programmation.

Abbes et Gargouri, 2018 [31], se sont concentrés sur la construction des ontologies pour l'intégration des Big Data. L'approche proposée est basée sur les bases de données NOSQL, à savoir MongoDB et les ontologies modulaires. Elle est divisée en trois (03) étapes consécutives : la première étape est l'enveloppement des sources de données dans les bases de données MongoDB (le contenu de chaque source de données est converti en une base de données MongoDB), la deuxième étape est la mise en correspondance des bases de données MongoDB avec les modules d'ontologie (chaque base de données MongoDB est mise en correspondance à un module d'ontologie OWL en utilisant des règles de transformation), la troisième étape est la fusion des modules d'ontologies pour en obtenir un global (les modules obtenus à l'étape précédente sont fusionnés afin d'obtenir une ontologie globale).
La principale innovation de l'approche proposée réside dans sa capacité de pouvoir être appliquée dans tous les domaines et dans sa dynamique. En effet, il est facile d'intégrer une nouvelle source de Big Data dans le processus d'intégration, puisque les sources de données sont traitées indépendamment les unes des autres. Ainsi, avec l'avènement d'une nouvelle source de données, la base de données MongoDB correspondante est mise en place et par conséquent son module d'ontologie correspondant, qui sera par la suite fusionné à l'ontologie globale. Selon les auteurs, la proposition constitue une base solide pour la construction d'une ontologie pour l'intégration des données massives. Néanmoins, il est certain que quelques aspects doivent être examinés avant et que d'autres doivent être étudiés de près. Cependant, les auteurs ont affirmé avoir établi un processus de construction d'ontologie, qui est facile à étendre et à mettre à jour si nécessaire.

Sottovia et al. 2019 [32], ont introduit une approche pour décrire comment le pipeline d'intégration des données a été mis en œuvre dans le projet Research Alps.
Le projet Research Alps vise à construire un ensemble de données décrivant des laboratoires de recherche, en supprimant les entités dupliquées appartenant à différentes classes d'entités et à fournir un scénario réel intéressant dans lequel ils peuvent expérimenter des techniques d'intégration de données, de correspondance d'entités et de fusion de données à grande échelle, en utilisant des sources de données hétérogènes en termes de schéma, de contenu, de format (elles traitent des bases de données structurées, CSV (Comma Separated Values) et relationnelles, et de sites web explorés) et de qualité (fréquence de mise à jour, ...). La proposition comprend trois (03) étapes principales : la première étape est l'extraction des entités, qui vise à identifier les données d'intérêt à partir de chaque source de données d'entrée et à les transformer en un certain nombre d'entités à l'aide d'un processus d'alignement des schémas effectué par un composant logiciel appelé "importer" qui leur permet d'accomplir cette tâche. La deuxième étape est l'appariement (couplage) des entités, qui consiste à dupliquer les entités brutes de la base de données à l'aide d'un algorithme appelé M-STEP. La troisième étape est la fusion des données, qui consiste à appliquer une stratégie de fusion permettant de concilier les conflits éventuels avec les appariements d'entités produits par l'étape précédente.

Giuseppe et Aversano, 2020 [33], ont proposé une approche qui permet un accès unifié à des données sources hétérogènes et indépendantes, offrant une approche d'intégration des données avec une architecture de médiation adoptée, qui créent une vue virtuelle des données réelles et permettent à des applications externes d'accéder aux données à travers cette vue de manière transparente. La transparence est garantie par la traduction des requêtes posées sur la vue virtuelle en requêtes directement exécutables à partir de sources locales. L'approche proposée permet un accès unifié à des sources hétérogènes grâce aux activités suivantes : recouvrement des sources, correspondance des schémas, fusion des schémas et enfin, reformulation des requêtes. Dans l'ensemble, l'approche est semi-automatique, mais par rapport aux systèmes existants, l'effort de l'utilisateur est réduit au minimum car il n'intervient que dans l'activité de configuration de la correspondance, en fixant le seuil des valeurs pour la génération et la validation des mappages. Des mappages simples

(1 : 1) et complexes (1 : n, n : 1 et n : m) sont générés. L'approche décrite est soutenue par un système logiciel spécialement conçu et développé. Le système fournit un premier niveau d'abstraction des activités et des composants impliqués dans leur exécution et un second niveau de spécialisation des composants. Bien que la conception du système vise à couvrir tous les aspects de l'intégration des données décrits jusqu'à présent, la mise en œuvre présente certaines limites. En particulier, l'acquisition de sources non structurées n'est pas encore envisagée dans le développement et le processus de rapprochement des données nécessite le développement des composants appropriés. À l'exception de ces activités, les processus d'intégration et de médiation sont entièrement pris en charge par le système.

Les tableaux ci-dessous résument les principales caractèristiques des approches citées ci-dessus. Les tableaux contiennent neuf (09) colonnes qui indiquent un critère de comparaison comme suit :

- La colonne **"Approche"** désigne l'approche de chaque article synthétisé.
- La colonne **"Degré d'automatisation"** précise le niveau d'automatisation de l'approche.
- La colonne **"Dataset"** indique les sources de données utilisées pour générer les recommandations.
- La colonne **"output"** indique la production finale de l'approche.
- La colonne **"Technique utilisée"** indique les méthodes utilisées pour la recommandation.
- La colonne **"Implémentation"** indique si l'approche a été implémentée.
- La colonne **"Avantage"** présente les principaux avantages de l'approche.
- La colonne **"Modèle sémantique"** indique le modèle sémantique utilisé pour l'intégration des données.
- La colonne **"Caractéristique de l'intégration"** indique le but dans lequel l'intégration est effectuée.

Approache	**Degré d'automatisation**	**Dataset**	**Output**	**Thechnique utilisée**	**Implémentation**	**Avantages**	**Modèle sémantique**	**Caractéristique de l'intégration**
Curé et al. 2013	*Manuel	*Bases de données NoSQL	*Framework pour l'intégration des données	*Analyse d'un ensemble de bases de données NoSQL pour générer des ontologies locales. *Génération d'une ontologie globale basée sur la découverte de correspondances entre les ontologies locales. *Proposition d'une solution de traduction de requêtes de SPARQL en langages d'interrogation des sources.	*Non	*Traitement possible même avec l'absence des schémas sources et de l'ontologie globale	*Ontologie	*Hétérogénéité des modèles de données. *Hétérogénéité des schémas. *Hétérogénéité des langages de requête.

TABLE 4.1 – Etat de l'art du travail connexe 1.

Approache	**Degré d'automatisation**	**Dataset**	**Output**	**Thechnique utilisée**	**Implémentation**	**Avantages**	**Modèle sémantique**	**Caractéristique de l'intégration**
Ajani, 2014	*Semi-automatique.	*Grand domaine de recensement (données semi-structurées et non structurées).	*Métadonnées sémantiques et une ontologie.	*Construction de l'ontologie du domaine de recensement. *Affectation d'une recherche sémantique.	*Oui	*Le modèle de données sémantiques basé sur la technologie du web sémantique sera prometteur pour fournir une meilleure solution en termes de coûts et d'avantages.	*XML et les ontologies OWL et RDF.	*Il est nécessaire d'améliorer la recherche d'informations sur les données de recensement des formulaires de recherche. Pour répondre à ce besoin, une approche basée sur une ontologie est proposée.

TABLE 4.2 – Etat de l'art du travail connexe 2.

Approache	**Degré d'automatisation**	**Dataset**	**Output**	**Thechnique utilisée**	**Implémentation**	**Avantages**	**Modèle sémantique**	**Caractéristique de l'intégration**
Jirkovský et Obitko, 2014	*Semi-automatique.	*Ontologie partagée. *Stockage partagé.	*Le système vise à fournir une solution de bout en bout, où les utilisateurs finaux formuleront des requêtes basées sur l'ontologie.	*Prétraitement des données. *Création d'une ontologie partagée. *Transformation des données.	*Oui	*La capacité de traiter diverses sources de données stockées dans des bases de données ou des fichiers ainsi que divers flux de données. Cette capacité permet de saisir avec précision les différentes relations entre les sources de données et peut donc améliorer le processus de prise de décision.	*Ontologie OWL.	*Construire une ontologie supérieure décrivant toutes les sources de données. *Transformer des données selon cette ontologie. *Analyser à l'aide du paradigme Big Data.

Table 4.3 – Etat de l'art du travail connexe 3.

Approache	**Degré d'automatisation**	**Dataset**	**Output**	**Thechnique utilisée**	**Implémentation**	**Avantages**	**Modèle sémantique**	**Caractéristique de l'intégration**
Bansal et kagemann ,2015	*Manuel.	*Les données disponibles sur le Web (données du gouvernement américain provenant de l'enquête nationale sur les voyages des ménages de la Federal Highway Administration...).	*Un data mart ou un entrepôt de données au service des Big Data.	*Génération des modèles sémantiques des ensembles de données en cours d'intégration. *Génération de données liées de manière sémantique conformément au modèle de données.	*Oui	*Le framework sémantique ETL proposé a un potentiel significatif pour produire des données liées qui soutiennent des applications innovantes basées sur les données pour un smart living.	*Ontologie OWL.	*Le processus sémantique ETL pourrait être introduit dans des outils logiciels ETL à source ouverte tels que CloverETL, une bibliothèque de moteurs Java qui n'a pas de composants d'interface utilisateur.

TABLE 4.4 – Etat de l'art du travail connexe 4.

Approache	**Degré d'automatisation**	**Dataset**	**Output**	**Thechnique utilisée**	**Implémentation**	**Avantages**	**Modèle sémantique**	**Caractéristique de l'intégration**
Knoblock et Szekley, 2015	*Automatique.	*Bases de données relationnelles. *Services web. *Excel, XML, JSON et CSV.	*Modèle sémantique.	*Importation. *Nettoyage. *Modélisation. *Intégration de données.	*Oui	*Analyser les descriptions de sources connues afin de proposer des mappages qui saisissent mieux la sémantique.	*Mappage des schémas.	*Capacité à importer diverses sources de données, y compris des sources relationnelles et hiérarchiques. *Possibilité d'intégrer facilement les données de différentes sources et de publier les résultats dans divers formats.

TABLE 4.5 – Etat de l'art du travail connexe 5.

Approache	**Degré d'automatisation**	**Dataset**	**Output**	**Thechnique utilisée**	**Implémentation**	**Avantages**	**Modèle sémantique**	**Caractéristique de l'intégration**
Keller et al. 2016	*Automatique.	*Sherlock. *Données brutes. *Fichiers.	*Triple RDF.	*Etablissement une description commune des données de l'ATM en utilisant un modèle d'ontologie. *Application des techniques d'intégration sémantique pour peupler un triplestore avec des données de Sherlock.	*Oui	*Le potentiel de servir une large communauté des chercheurs d'aviation, le personnel de gestion des vols opérationnels, les décideurs politiques, et au-delà.	*Ontologie.	*Permettre des découvertes qui pourraient éventuellement avoir un impact sur l'efficacité et la sécurité de notre système de transport aérien.

TABLE 4.6 – Etat de l'art du travail connexe 6.

Approache	**Degré d'automatisation**	**Dataset**	**Output**	**Thechnique utilisée**	**Implémentation**	**Avantages**	**Modèle sémantique**	**Caractéristique de l'intégration**
Fang et al. 2016	*Semi-automatique.	*Métadonnées (MARC, DC, GILS).	*Interface.	*Collecte de données. *Sauvegarde des données. *Traitement des données. *Fourniture d'informations. *Maintenance et mise à jour des informations.	*Oui	*Aider à poser les bonnes bases pour réaliser une gestion intensive, intelligente et unifiée du tourisme.	*Ontologie RDF.	*Résoudre le problème de l'uniformité et de l'interopérabilité d'une source de données hétérogène.

TABLE 4.7 – Etat de l'art du travail connexe 7.

Approache	**Degré d'automatisation**	**Dataset**	**Output**	**Thechnique utilisée**	**Implémentation**	**Avantages**	**Modèle sémantique**	**Caractéristique de l'intégration**
Abbes et Gargouri, 2017	*Automatique.	*Données structurées. *Données semi-structurées. *Données non-structurées.	*Ontologie.	*Enveloppement des sources de données dans des bases de données MongoDB. *Construction de l'ontologie locale. *Construction de l'ontologie globale à partir des ontologies locales.	*Oui	*Les ontologies peuvent être utiles pour la réutilisation dans le domaine de la représentation des connaissances.	*Ontologie OWL.	*Hétérogénéité des modèles de données. *Hétérogénéité des schémas.

TABLE 4.8 – Etat de l'art du travail connexe 8.

Approache	**Degré d'automatisation**	**Dataset**	**Output**	**Thechnique utilisée**	**Implémentation**	**Avantages**	**Modèle sémantique**	**Caractéristique de l'intégration**
Vathy-Fogarassy et Hugyák, 2017	*Semi-automatique.	*Données structurées. *Données semi-structurées.	*Méthode d'intégration des données.	*Schéma général. *Adaptateurs pour bases de données. *Migration de données provenant de différents systèmes sources. *Stratégie de traitement des demandes.	*Oui	*Présenter une nouvelle méthode d'intégration des données et un nouveau flux de travail, qui mettent en œuvre l'intégration des données de différentes sources de manière à ce que les utilisateurs n'aient pas besoin de compétences en programmation.	*NoSQL.	*Hétérogénéité des modèles de données. *Hétérogénéité des schémas. *Hétérogénéité des langages de requête.

TABLE 4.9 – Etat de l'art du travail connexe 9.

Approache	**Degré d'automatisation**	**Dataset**	**Output**	**Thechnique utilisée**	**Implémentation**	**Avantages**	**Modèle sémantique**	**Caractéristique de l'intégration**
Abbes et Gargouri, 2018	*Semi-automatique.	*Modules d'ontologie.	*Ontologie globale.	*Détection des chevauchements entre les deux modules d'ontologie comparés. *Calcul des mesures de similarité. *Mise à jour d'un module avec les concepts, les attributs et les relations de l'autre module.	*Oui	* Eviter la redondance dans l'ontologie globale sans perte d'information.	*Ontologie OWL.	*Proposer une nouvelle approche pour composer automatiquement des modules d'ontologie.

TABLE 4.10 – Etat de l'art du travail connexe 10.

Approache	**Degré d'automatisation**	**Dataset**	**Output**	**Thechnique utilisée**	**Implémentation**	**Avantages**	**Modèle sémantique**	**Caractéristique de l'intégration**
Sottovia et al. 2019	*Automatique.	*CSV. *Bases de données relationnelles. *Sites web explorés.	*Ensemble de données avec suppression des entités dupliquées appartenant à des classes d'entités différentes.	*Correspondence. *Fusion.	*Oui	*Efficacité. *Robustesse aux valeurs manquantes. *Traitement des valeurs manquantes. *Préservation des relations hiérarchiques entre les données.	*Mappage d'entités.	*Offrir un scénario réel intéressant où il est possible d'expérimenter des techniques d'intégration de grandes données, d'appariement d'entités et de fusion de données.

TABLE 4.11 – Etat de l'art du travail connexe 11.

Approache	**Degré d'automatisation**	**Dataset**	**Output**	**Thechnique utilisée**	**Implémentation**	**Avantages**	**Modèle sémantique**	**Caractéristique de l'intégration**
Giuseppe et Aversano, 2020	*Semi-automatique.	*Données structurées. *Données semi-structurées. *Données non-structurées.	*Ontologie globale.	*Enveloppement des sources de données. *Appariement des schémas. *Fusion des schémas. *Reformulation des requêtes.	*Oui	*La transparence est garantie par la traduction des requêtes posées sur la vue virtuelle en requêtes directement exécutables à partir de sources locales.	*Ontologie OWL.	*Permettre un accès unifié à des données sources hétérogènes et indépendantes, offrant une approche d'intégration des données.

TABLE 4.12 – Etat de l'art du travail connexe 12.

Les douze (12) tableaux présentent les principales caractéristiques des travaux les plus liés à notre recherche. Ces caractéristiques consistent à préciser si les approches existantes se concentrent sur le data mining, sont orientées vers les Big Data, s'alignent sur d'autres ontologies, utilisent OWL/RDF dans le modèle sémantique, le tableau présente également, l'objectif de chaque approche.

4.3 Analyse comparative :

Les travaux existants tentent de résoudre les problèmes d'intégration des données sous différents angles. Nous résumons dans cette section la manière dont les problèmes d'intégration de données peuvent être résolus en se basant sur les solutions présentées précédemment.

- **Résolution de l'hétérogénéité des modèles de données :** l'hétérogénéité des modèles de données peut être résolue en utilisant un schéma global des systèmes intégrés. Le schéma global permet de fournir une représentation unifiée des données et il peut être représenté en utilisant un format standard tel que : JSON ou XML. La tâche principale dans la définition du schéma global est la mise en correspondance entre les schémas sources et le schéma global.

- **Résolution de l'hétérogénéité des schémas :** une approche basée sur l'ontologie peut être une solution efficace à l'hétérogénéité sémantique. Elle fournit un vocabulaire commun sur un domaine spécifique en utilisant un ensemble de concepts et les relations entre eux. Cela peut aider en générant des entités correspondant et intégrant des schémas de différentes sources.

- **Résolution de l'hétérogénéité des langages d'interrogation :** un langage d'interrogation unifié est nécessaire afin de résoudre l'hétérogénéité des langages d'interrogation des systèmes intégrés. Ce langage d'interrogation doit être puissant dans le sens où il doit mettre en œuvre des requêtes simples et complexes. En plus du langage d'interrogation unifié, un ensemble de pilotes qui traduit le langage d'interrogation unifié dans les langages d'interrogation des systèmes intégrés est également nécessaire.

Toutefois, l'analyse des travaux connexes a révélé quelques problèmes critiques, qui ne sont pas résolus dans les technologies existantes :

- **L'absence de solution pour l'intégration avec des réponses en temps réel :** de nombreuses solutions existantes sont basées sur le cadre Hadoop-MapReduce, qui a principalement résolu le problème du volume de données. Cependant, en raison de l'algorithme de tri extravagant sur lequel Hadoop s'appuie fortement pour exécuter la fonction de réduction, la performance peut être un goulot d'étranglement.

- **L'absence de solution pour l'intégration avec la prise en compte de la qualité des données :** l'intégration des données médicales est confrontée à un problème de qualité. L'intégration de données hétérogènes peut produire des données désordonnées ou non significatives. C'est pourquoi les solutions d'intégration doivent résoudre les problèmes de qualité. Malheureusement, nous n'avons pas trouvé de solution qui puisse résoudre efficacement ce problème.

- **Le manque de solution pour l'intégration avec la prise en compte de toutes les sources de données :** en général, peu d'approches offrent l'intégration de données provenant des sources de données structurées, semi-structurées et non structurées.

En résumé, les Big Data manquent encore d'une approche globale et solide de la gestion des données. La gestion de l'information dans le contexte des technologies des données massives est un sujet essentiel pour la recherche future. Un défi qui implique l'ensemble du processus du pipeline Big Data, c'est-à-dire l'ensemble des tâches requises pour piloter les calculs des grandes données. La documentation, la reconfiguration, l'assurance qualité des données et la vérification sont des exemples de tâches cruciales qui ne sont pas facilement prises en charge dans les technologies Big Data.

Nous proposons une approche améliorée pour l'intégration des données qui résume les travaux cités ci-dessus et permet un accès rapide et facile aux données grâce à une vue globale utilisant le SPARQL, qui est un langage de requête et un protocole qui permet de

rechercher, d'ajouter, de modifier ou de supprimer des données RDF disponibles à travers Internet, c'est l'une des technologies clés du Web sémantique présentée par W3C.

4.4 Conclusion :

Dans ce chapitre, nous avons élaboré l'état de l'art qui représente tous les travaux connexes que nous avons synthétisés, nous avons présenté ceci dans un tableau qui contient les grandes lignes de chaque document synthétisé, tout en suivant chaque travail par un bref paragraphe qui le résume, par la suite nous avons procédé à une analyse comparative entre les approches des documents connexes et notre approche.

Dans le chapitre suivant, nous détaillerons notre approche et ses différentes étapes.

Chapitre 5

Approche

Sommaire

5.1 Introduction :

Dans ce chapitre, nous présenterons en détail notre approche d'intégration de données dans un contexte Big Data utilisant les ontologies comme modèle sémantique.

5.2 Etapes de l'approche :

La recherche bibliographique que nous avons menée, a révélé que le travail de recherche sur l'extraction des connaissances des bases de données est basé sur des données struc-

turées, c'est-à-dire des données relationnelles. Cependant, peu d'ouvrages traitement les données semi-structurées ou non-structurées. Nous proposons donc une approche pour extraire les connaissances des Big Data pour l'amélioration de la qualité de ces données massives.

La **figure 5.1** donne un aperçu du système proposé qui se compose de cinq (05) étapes intitulées comme suit : collecte des données qui consiste à importer des données hétérogènes de sources disparates, nettoyage des données qui consiste à filtrer les données hétérogènes importées en supprimant toute donnée dupliquée, manquante ou aberrante, analyse des données qui consiste à charger chaque source de données nettoyées dans des bases de données NoSQL (MongoDB dans notre cas), construire l'ontologie locale pour chaque source et enfin jumeler toutes les ontologies locales pour en construire une globale, enrichissement de l'ontologie qui consiste à comparer l'ontologie globale résultante avec une ontologie existante, large et du même domaine afin d'améliorer sémantiquement notre ontologie globale et enfin, construction des requêtes qui consiste à l'interrogation de l'ontologie globale par l'utilisateur avec des requêtes SPARQL.

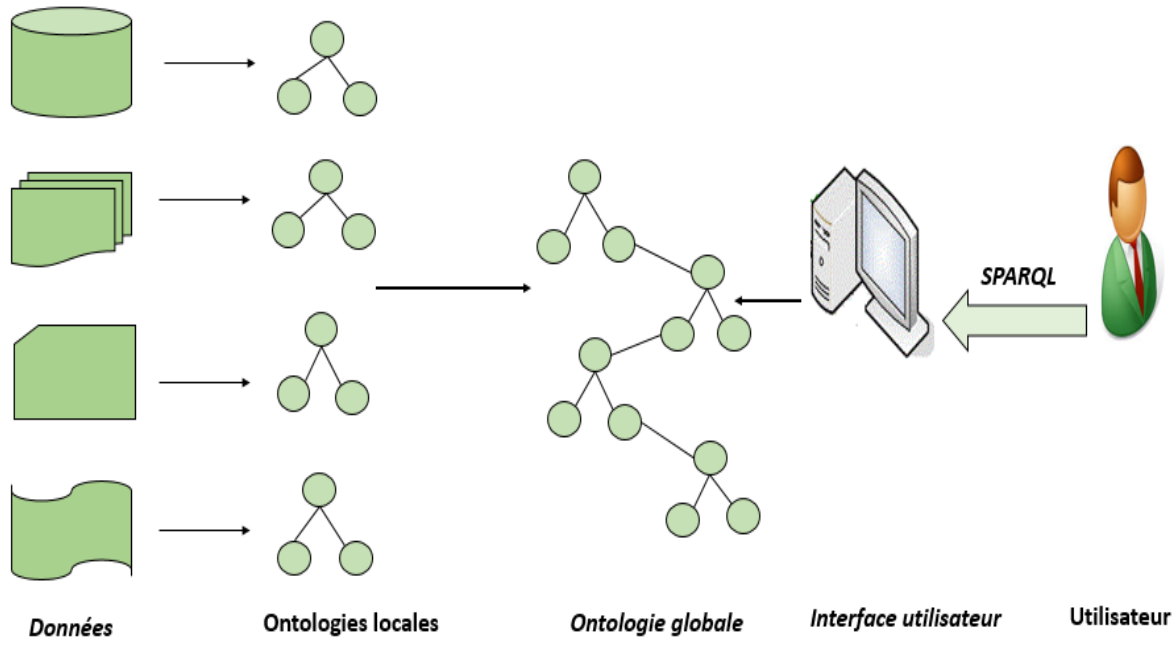


FIGURE 5.1 – Schéma global de l'approche.

Nous détaillons ci-dessous chaque étapes comme suit :

5.2.1 Collecte des données :

Cette partie consiste à collecter des informations, dans les logiciels, cette étape s'appelle le chargement des ensembles de données. Elle permet l'acquisition des informations nécessaires à la génération de l'ontologie (concepts, attributs, relations et axiomes) à partir d'une source de données existante, ces sources d'entrée peuvent être de plusieurs types : structurées, semi-structurées ou non structurées.

L'objectif le plus important de la collecte de données est de s'assurer que des données riches en informations et fiables sont collectées pour l'analyse statistique afin que des décisions fondées sur les données puissent être prises pour la recherche.

Il s'agit d'une composante intégrale, généralement initiale, de toute recherche effectuée dans un domaine d'étude tel que les sciences physiques et sociales, les affaires, les sciences humaines et autres.

5.2.2 Nettoyage des données :

Le nettoyage des données est le processus de préparation des données en vue de leur analyse par la suppression ou la modification des données qui sont incorrectes, incomplètes, non pertinentes, dupliquées ou mal formatées. Ces données ne sont généralement pas nécessaires ou utiles lorsqu'il s'agit d'analyser des données car elles peuvent entraver le processus ou fournir des résultats inexacts. Il existe plusieurs méthodes pour nettoyer les données en fonction de la manière dont elles sont stockées et des réponses recherchées.

Le nettoyage des données joue un rôle essentiel dans le processus ETL utilisé pour l'extraction et la compilation des données brutes pour les transformer en données intelligibles et les charger dans un système cible, tel qu'une base de données ou un entrepôt de données pour un accès et une analyse faciles, ce processus permet de garantir la cohérence, l'exactitude et la qualité des informations.

5.2.3 Analyse des données :

Dans un environnement de données massives, les données proviennent de diverses sources. Les bases de données NoSQL sont utilisées comme des bases de données.

Ces bases de données souffrent d'un manque de sémantique. Afin de résoudre le problème, une approche basée sur l'ontologie est proposée pour extraire la sémantique cachée des grandes données. L'objectif du système proposé est de construire plus tard une ontologie globale pour les grandes données provenant de différentes sources de données.

Initialement, des ontologies locales doivent être construites pour chaque source de données. Le système proposé traite les bases de données MongoDB comme des données d'entrée et implique ces trois (03) étapes : homogénéisation des données, définition des règles de mapping et construction de l'ontologie globale. Nous expliquons chaque étape comme suit :

5.2.3.1 Homogénéisation des données :

Dans la première étape, nous enveloppons chaque source de données dans une base de données MongoDB en tant que formalisme de représentation intermédiaire. Pour charger les données dans les bases de données MongoDB, nous choisissons d'utiliser l'outil d'intégration de données "Talend for Big Data" [39]. Cet outil permet d'extraire des données de sources de données importantes et hétérogènes et de les intégrer dans des bases de données NoSQL.

5.2.3.2 Définition des règles de mapping :

Pour générer de la sémantique à partir du SGBD "MongoDB", celui-ci est mis en correspondance avec l'ontologie OWL grâce aux règles suivantes :

- **Création du squelette de l'ontologie :** dans cette étape, nous extrayons les classes de l'ontologie des collections ainsi que les relations de subsomption afin de définir l'arborescence de l'ontologie.

Règle une : chaque collection de la base de données doit être transformée en une classe d'ontologie.

Règle deux : une relation de subsomption R (parent/enfant) est extraite du champ "parent" dans chaque document. R doit être transformée en une relation "subClassOf" dans l'ontologie.

- **Identification des propriétés :** chaque document d'une collection est composé d'un ensemble de paires (champ/valeur). Les champs peuvent avoir différents types, à savoir des types de base (string, int, boolean ...).

  Règle trois : chaque champ de base d'un document est transformé en "dataTypeProperty" dans la classe correspondant à la collection contenant ce document dans l'ontologie.

- **Apprentissage des individus (instances et faits) :**

  Règle quatre : les valeurs des champs dans les documents sont transformées en individus dans l'ontologie.

- **Déduction des axiomes :** le langage OWL propose des constructions pour définir des axiomes de classe tels que "owl :equivalentClass" et "owl :disjointWith".

L'équivalence : deux concepts sont équivalents s'ils ont la même extension.

La disjonction (l'incompatibilité) : deux concepts sont équivalents s'ils ont la même extension.

Nous considérons deux classes Cl et C2 et 11, 12 sont des individus de Cl et C2 respectivement.

Règle cinq : Cl est équivalent à C2 si les deux extensions de la classe contiennent exactement le même ensemble d'individus, sinon Cl est disjoint de C2.

Les modules obtenus à l'étape précédente sont fusionnés afin d'obtenir l'ontologie locale correspondant à la source de données.

5.2.3.3 Construction de l'ontologie globale :

Il s'agit de la création d'une nouvelle ontologie à partir de deux ou plusieurs ontologies existantes du même domaine ou de domaines connexes, la nouvelle ontologie est censée contenir les connaissances des ontologies initiales.

La **figure 5.2** représente un passage des ontologies locales à une ontologie globale.

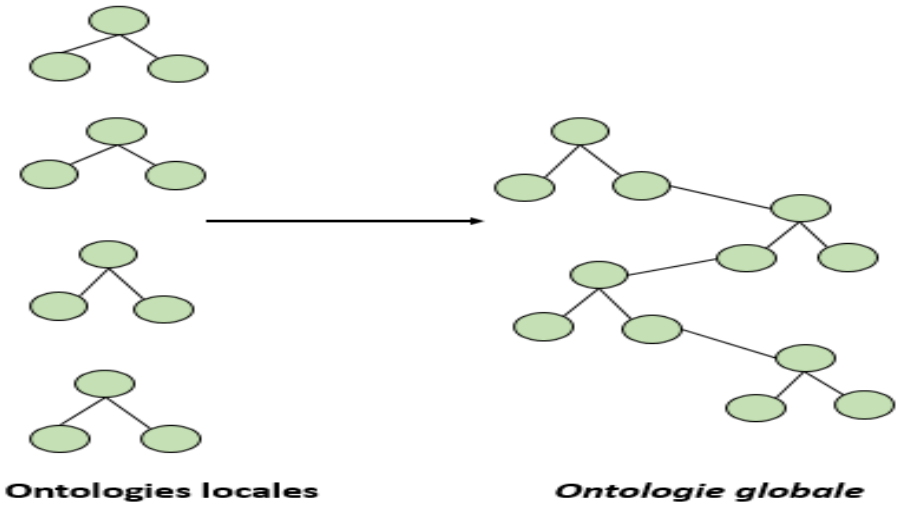


FIGURE 5.2 – Passage des ontologies locales à une ontologie globale.

De nombreuses ontologies font référence au même domaine et aux mêmes objets, il est de plus en plus nécessaire de les fusionner et de les organiser. En effet, le but ultime de la fusion est de représenter une meilleure perspective de la connaissance d'un domaine.

En général, la fusion d'ontologies est utilisée dans le domaine de l'intégration des données, mais elle peut également être considérée comme une technique utilisée dans le domaine de l'enrichissement des ontologies, que nous détaillerons plus tard, elle consiste

à insérer les connaissances connexes dans l'ontologie en moins de temps et de coût.

La manière dont le processus de fusion est réalisé est encore très peu claire. En effet, il n'y a pas de consensus sur la méthodologie de fusion des ontologies. La seule phase commune est la phase initiale qui prend un ensemble d'ontologies (deux ou plus) comme entrée. Certaines commencent directement avec toutes les ontologies à fusionner (méthode non incrémentale), d'autres commencent avec un groupe initial sélectionné d'ontologies (généralement une ontologie) qui est ensuite progressivement élargi par les autres ontologies (méthode incrémentale).

Enfin, l'ontologie construite doit être vérifiée comme une ressource correcte, cohérente et complète de la connaissance spécifique du domaine. La vérification doit être faite en terme de structure de l'ontologie et de son contenu sous-jacent. À cette fin, nous déployons le framework "Protégé" pour valider l'ontologie produite, pour vérifier si elle est logiquement cohérente et pour générer un fichier OWL approprié.

5.2.4 Enrichissement de l'ontologie globale :

L'enrichissement est le processus qui cherche de nouvelles entités (généralement à partir de ressources textuelles externes) et les place correctement au sein de l'ontologie à enrichir (dans notre cas, l'ontologie globale).

Les ontologies jouent un rôle important pour de nombreux projets à forte intensité de connaissances pour lesquelles elles fournissent une source de termes précisément définis. Toutefois, leur utilisation généralisée pose des problèmes de prolifération. Les ingénieurs ou les utilisateurs de l'ontologie ont souvent une ontologie de base qu'ils utilisent, par exemple, pour la navigation ou l'interrogation de données, cependant, ils doivent l'étendre avec, l'adapter ou la comparer avec d'autres ontologies existantes plus larges. Pour détecter et récupérer les ontologies pertinentes pour l'enrichissement, il faut disposer de moyens permettant de mesurer la similitude entre les ontologies.

- **Calcul de similarité :** La question de l'identification des similitudes et/ou du calcul des distances sémantiques est considérée comme un sujet de recherche très étudié dans les domaines de l'informatique, de l'intelligence artificielle et de la linguistique.

En particulier, le domaine de la recherche d'informations qui repose largement sur les mesures d'identification de similitude entre documents.

Le problème de ces approches est qu'elles se concentrent généralement sur les mots uniques d'un document en ignorant les relations ontologiques qui existent entre les mots. Nous pouvons distinguer trois (03) façons de déterminer la similarité sémantique entre les objets dans l'ontologie :

La première approche indique l'évaluation de la similarité par le contenu de l'information (également appelée approche basée sur les nœuds).

La deuxième approche représente une évaluation de la similarité basée sur la distance conceptuelle (également appelée approche basée sur les bords).

La troisième approche est hybride et combine les deux premières approches.

Le premier type est basé sur les nœuds. Les travaux menés dans le cadre de cette approches ont utilisé le contenu typiquement basé sur l'information pour déterminer la similarité conceptuelle. De plus, la similarité entre deux concepts est obtenue par le degré de partage de l'information.

Le second type est basé uniquement sur la hiérarchie ou les distances aux bords. Le problème de cette approche est que les arcs de taxonomie qui sert à organiser hiérarchiquement les informations représentent des distances uniformes, c'est-à-dire que tous les liens sémantiques ont le même poids.

Enfin, l'approche hybride qui combine les deux approches présentées ci-dessus.

Cette comparaison révèle que la mesure de Wu et Palmer a l'avantage d'être simple à calculer, en plus des performances qu'elle présente tout en restant aussi expressive que les autres. C'est pourquoi nous avons adopté cette mesure comme base de notre travail.

- **La mesure de Wu et Palmer :** Le principe du calcul de similarité est basé sur la méthode de comptage des bords qui est définie comme suit : etant donné une ontologie formée par un ensemble de nœuds et un nœud racine (R) (Fig. 5.1). C1 et C2 représentent deux éléments de l'ontologie dont nous calculerons la similitude. Le principe du calcul de similarité est basé sur la distance (N1 et N2) qui sépare les nœuds C1 et C2 du nœud racine et la distance (N) qui sépare l'ancêtre commun (CS) le plus proche de C1 et C2 à partir du nœud R. La mesure de similarité de Wu et Palmer est définie par la l'expression suivante :
  SimWP=2*N/(N1+N2).

La **figure 5.3** représente un exemple d'hiérarchie des concepts.

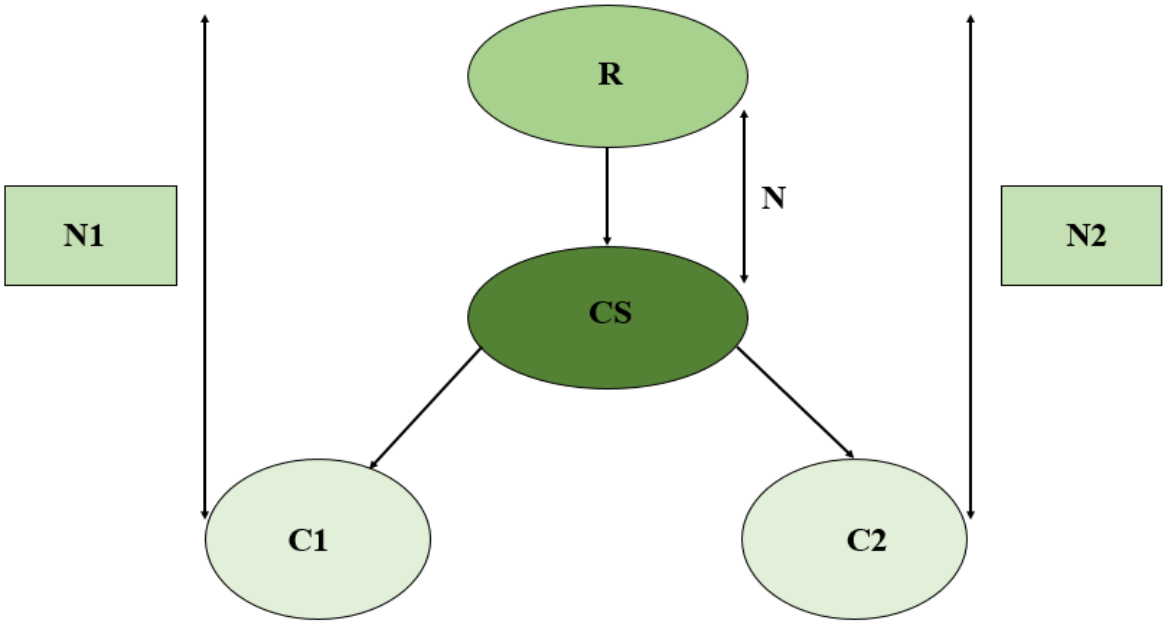


FIGURE 5.3 – Exemple de l'hiérarchie des concepts.

5.2.5 Construction des requêtes :

La réponse aux requêtes est importante dans le contexte du Web sémantique, car elle permet un mécanisme par lequel les utilisateurs et les applications peuvent interagir avec les ontologies et les données.

Plusieurs langages d'interrogation ont été conçus à cette fin, notamment RDQL(RDF Data Query Language), SeRQL(Structured Entity Relationship Query Language) et, plus récemment, SPARQL.

Nous considérons le langage d'interrogation SPARQL [43], qui a été normalisé en 2008 par le W3C et qui est maintenant soutenu par la plupart des magasins triple RDF.

La **figure 5.2** repésente un schéma global de notre approche.

5.3 Conclusion :

Dans ce chapitre, nous avons présenté en détail notre approche d'intégration de données dans un contexte Big Data utilisant les ontologies comme modèle sémantique.

Dans le chapitre suivant, nous procéderons à l'explication de tous les aspects liés à l'implémentation de notre approche.

Chapitre 6

Implémentation et tests

Sommaire

6.1 Introduction :

Dans ce chapitre, nous aborderons les divers aspects liés à l'implémentation des prototypes que nous avons développés, à savoir, les technologies, les logiciels choisis en utilisant differentes sources de données pour l'implémentation de notre approche.

6.2 Description des Datasets :

Il s'agit d'une collecte de données sur les tendances chronologiques avec une série de fichiers JSON et CSV, principalement axés sur les pays les plus touchés par la crise sanitaire COVID 19 (Corona Virus Disease 2019). Les données de séries chronologiques sous forme d'arbre doivent permettre de réaliser différents types d'analyses pour répondre aux questions sur ce qui peut rendre le système de santé d'un pays vulnérable à la crise du COVID 19 et sur les données démographiques relatives à la santé qui peuvent contribuer à réduire l'impact [34].

Dataset : un ensemble de données (ou jeu de données) est une collection de données. Dans le cas des données tabulaires, un ensemble de données correspond à une ou plusieurs tables de base de données, où chaque colonne d'une table représente une variable particulière, et chaque ligne correspond à un enregistrement donné de l'ensemble de données en question. L'ensemble de données énumère les valeurs de chacune des variables, telles que la taille et le poids d'un objet, pour chaque membre de l'ensemble de données. Chaque valeur est connue sous le nom de donnée. Les ensembles de données peuvent également consister en un ensemble de documents ou de fichiers [35].

6.3 Environnements de developpement :

6.3.1 Anaconda :

C'est une distribution libre et open source des langages de programmation Python et R, appliqué au développement d'applications dédiées à la science des données et à l'apprentissage automatique .

Une distribution est un langage de programmation, certaines librairies et autres fonctionnalités. Anaconda est donc une distribution Python, faite pour la Data Science [36].

6.3.2 Jupyter notebook :

C'est une interface Web dans laquelle nous pouvons taper du code Python, l'exécuter et voir directement les résultats, y compris une visualisation à l'aide de graphiques [36].

6.3.3 MongoDB :

C'est une base de données NoSQL orientée documents apparue vers le milieu des années 2000, elle est utilisée pour le stockage de gros volumes de données. Au lieu d'utiliser des tables et des lignes comme dans les bases de données relationnelles traditionnelles, MongoDB utilise des collections et des documents. Les documents sont constitués de paires clé-valeur qui constituent l'unité de base des données dans MongoDB. Les collections contiennent des ensembles de documents et fonctionnent comme l'équivalent des tables de bases de données relationnelles [37].

6.3.4 NoSQL :

En informatique et en bases de données, NoSQL est une famille de SGBD qui s'écarte du paradigme classique des bases de données relationnelles. L'explication la plus populaire de l'acronyme est "Not only SQL" qui signifie pas seulement SQL en anglais [38].

6.3.5 Talend for Big Data :

Talend Open Studio est une plate-forme d'intégration de données Open Source, basée sur le langage Java, de type ETL (Extract Transform Load) , développée par la société Talend.Il permet d'extraire des données d'une source, de modifier ces données, puis de les recharger vers une destination. La source et la destination des donnéees peuvent être une

base de données, un service web, un fichier csv et bien d'autres. . . etc [39].

6.3.5.1 Job Talend :

Un Job est la représentation graphique fonctionnelle d'un ou plusieurs composants connectés, permettant de définir et d'exécuter des processus de gestion de flux de données. En d'autres termes, le Job permet de mettre en place votre flux de données [39].

6.3.6 Protégé :

C'est une plateforme gratuite et open-source qui fournit à une communauté d'utilisateurs croissante une suite d'outils pour construire des modèles de domaine et des applications basées sur la connaissance avec des ontologies, il inclut de nombreux plugins pour la manipulation et la représentation d'ontologies dans différents formats [40].

6.4 Langage de programmation :

6.4.1 Python :

C'est un langage de programmation open source, interprété, qui ne nécessite pas d'être compilé pour fonctionner. Ceci permet de voir rapidement les résultats d'un changement dans le code. toutefois, ce langage de programmation s'est hissé parmi les plus utilisés dans le domaine du développement de logiciels, de gestion d'infrastructure et d'analyse de données. Il s'agit d'un élément moteur de l'explosion du Big Data [41].

6.4.2 JSON :

Acronyme de JavaScript Object Notation. Il s'agit d'un format de fichier open-standard permettant de stocker les données de manière organisée et lisible par un humain tout en y facilitant l'accès. Etroitement lié à JavaScript, ce format peut toutefois être généré et

lu par la plupart des langages de programmation. Cette universalité lui a permis de devenir une façon très populaire de stocker, organiser, lire et partager des données dans les applications et services web [42].

6.4.3 SPARQL :

Acronyme de SPARQL Protocol and RDF Query Language. C'est un langage de requête RDF, autrement dit, un langage de requête sémantique pour les bases de données, capable de récupérer et de manipuler des données stockées au format RDF [43].

6.5 Bibliothèques de Python :

6.5.1 Pandas :

Acronyme de Python Data Analysis Library.C'est une bibliothèque écrite pour le langage de programmation Python permettant la manipulation et l'analyse des données.

Pandas, prend des données (comme un fichier CSV ou TSV (Tab Separated values), ou une base de données SQL) et crée un objet Python avec des lignes et des colonnes appelées DataFrames.

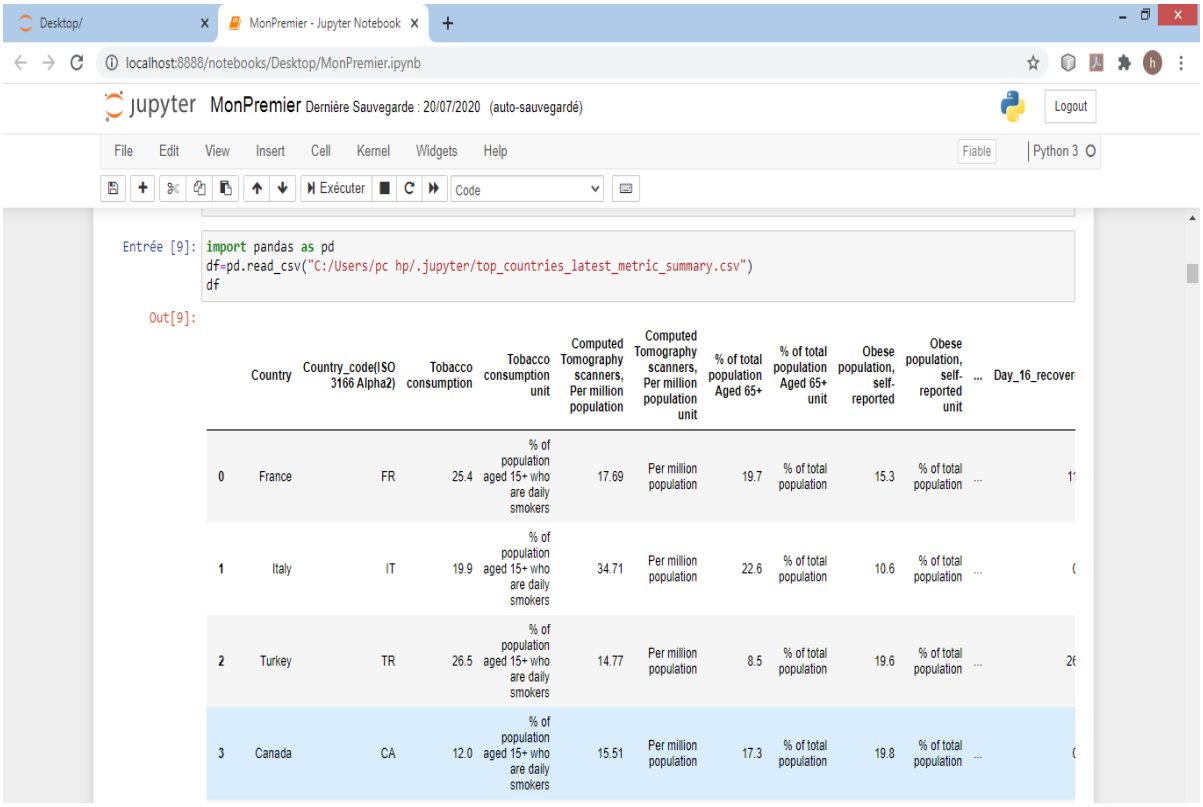
Cela fait des pandas un allié de confiance dans le domaine de la science des données et de l'apprentissage automatique [44].

6.5.2 Numpy :

C'est une extension du langage de programmation Python, destinée à manipuler des matrices ou tableaux multidimensionnels ainsi que des fonctions mathématiques opérant sur ces tableaux [45].

6.6 Mise en service :

La **figure 6.1** représente l'étape de la collecte des données, nous avons importé ces dataset hétérogènes du site "Kaggle", après l'mportation, les données sont sous forme d'un dataframe (structure de données de deux dimentions).



	Country	Country_code(ISO 3166 Alpha2)	Tobacco consumption	Tobacco consumption unit	Computed Tomography scanners, Per million population	Computed Tomography scanners, Per million population unit	% of total population Aged 65+	% of total population Aged 65+ unit	Obese population, self-reported	Obese population, self-reported unit	...	Day_16_recover
0	France	FR	25.4	% of population aged 15+ who are daily smokers	17.69	Per million population	19.7	% of total population	15.3	% of total population	...	1
1	Italy	IT	19.9	% of population aged 15+ who are daily smokers	34.71	Per million population	22.6	% of total population	10.6	% of total population	...	
2	Turkey	TR	26.5	% of population aged 15+ who are daily smokers	14.77	Per million population	8.5	% of total population	19.6	% of total population	...	2
3	Canada	CA	12.0	% of population aged 15+ who are daily smokers	15.51	Per million population	17.3	% of total population	19.8	% of total population	...	

FIGURE 6.1 – Collecte des données.

La **figure 6.2** représente les données avant le nettoyage, ces données contiennent des valeurs manquantes (nan) et dupliquées :

Remarque : l'unité est dupliquée, néanmoins, nous ne pouvons pas la supprimer car elle est au même temps informative.

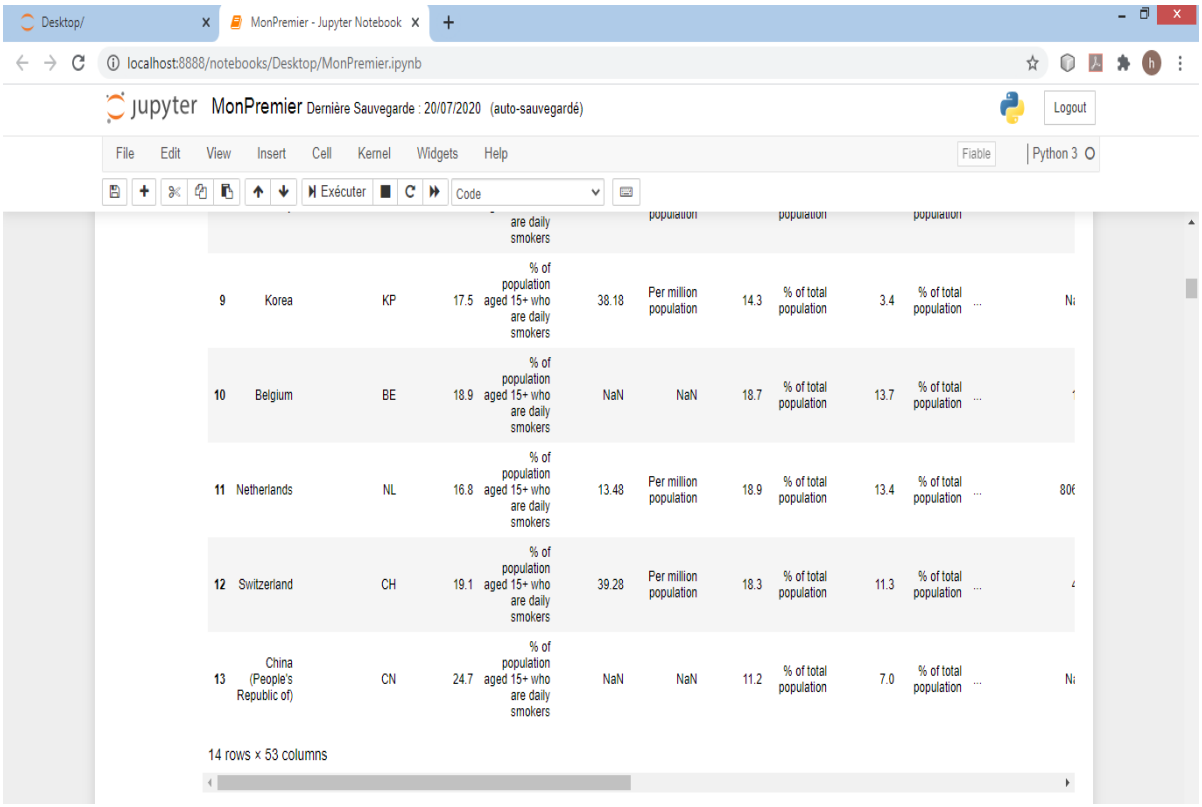


FIGURE 6.2 – Données avant le nettoyage.

La **figure 6.3** représente la méthode de traitement des données non numériques manquantes, grâce à la bibliothèque "Numpy" destinée à manipuler des tableaux multidimentionnels :

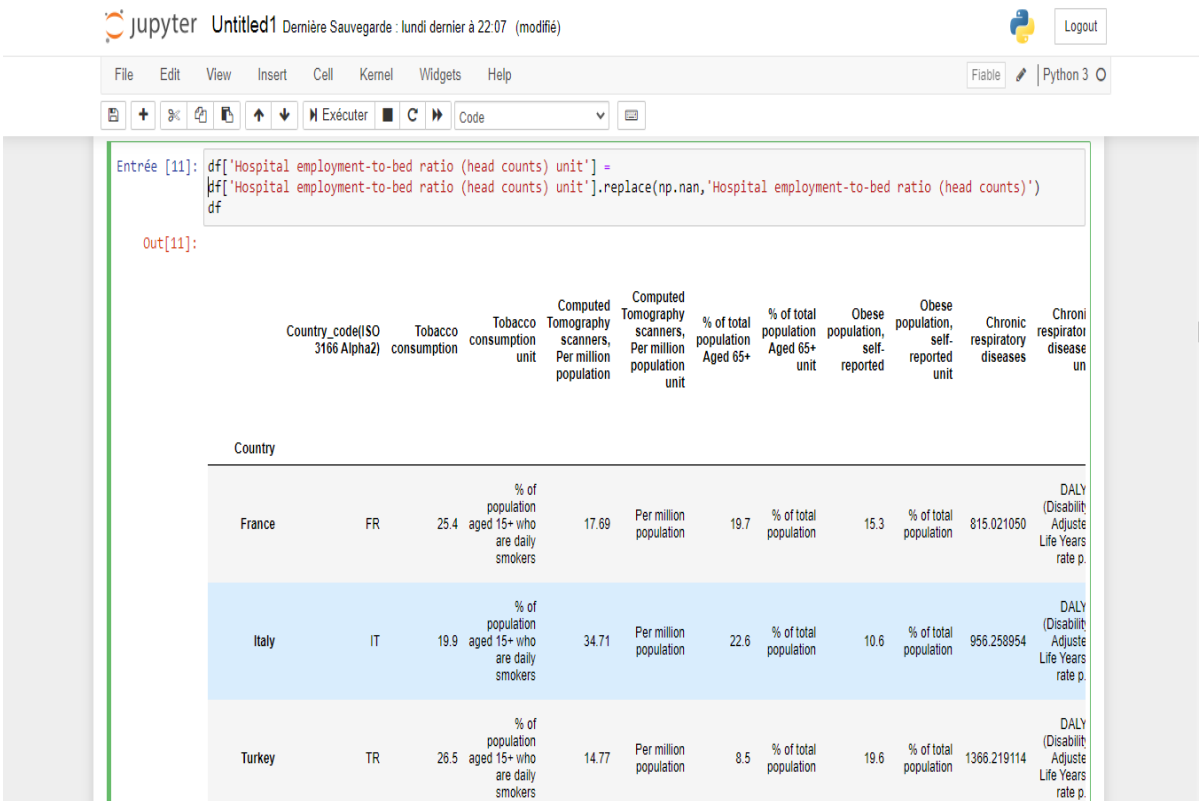


FIGURE 6.3 – Traitement des données non numériques manquantes.

La **figure 6.4** représente la méthode de traitement des données numériques manquantes, grâce à la fonction "fillna" qui permet de remplacer les données numériques par la valeur zéro (0) :

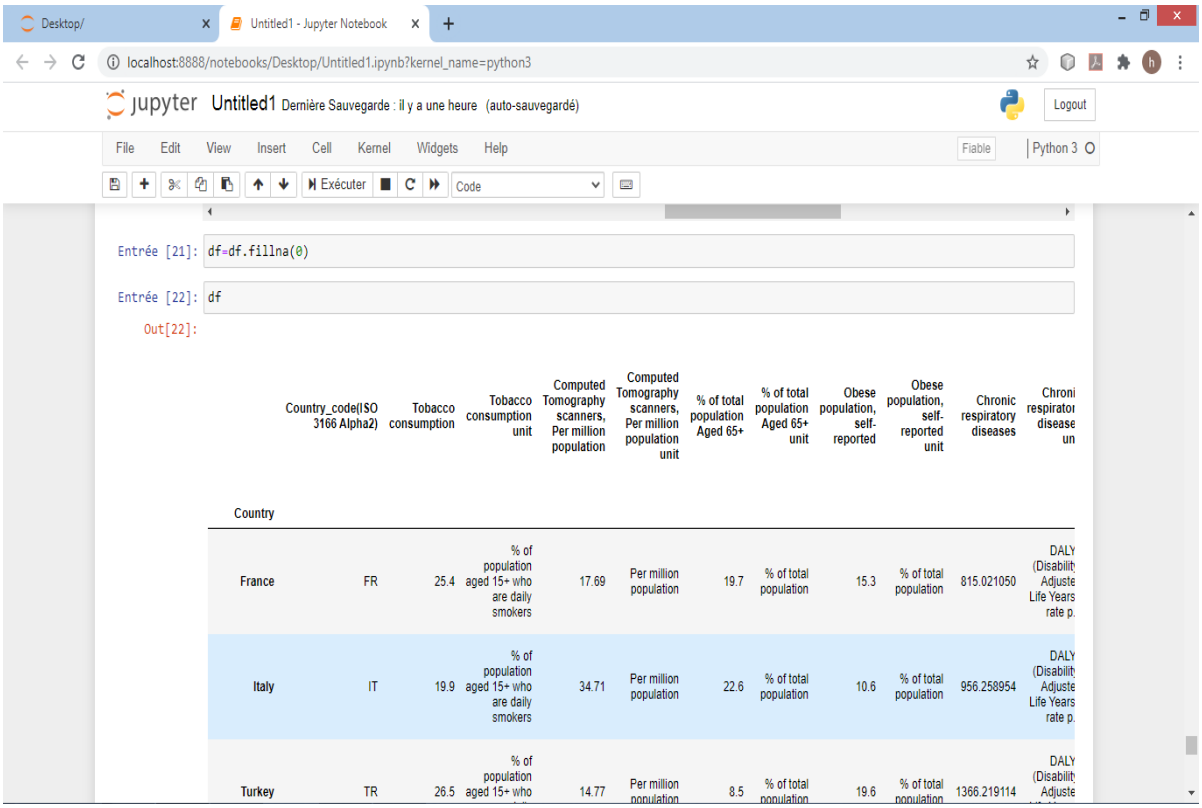


FIGURE 6.4 – Traitement des données numériques manquantes.

La **figure 6.5** représente les données aprés le nettoyage, toutes les données manquantes ont été remplacées par la valeur zéro (0) dans le cas des données numériques et par la valeur qui lui convient dans le cas des données non numériques (la valeur de la colonne correspondante) :

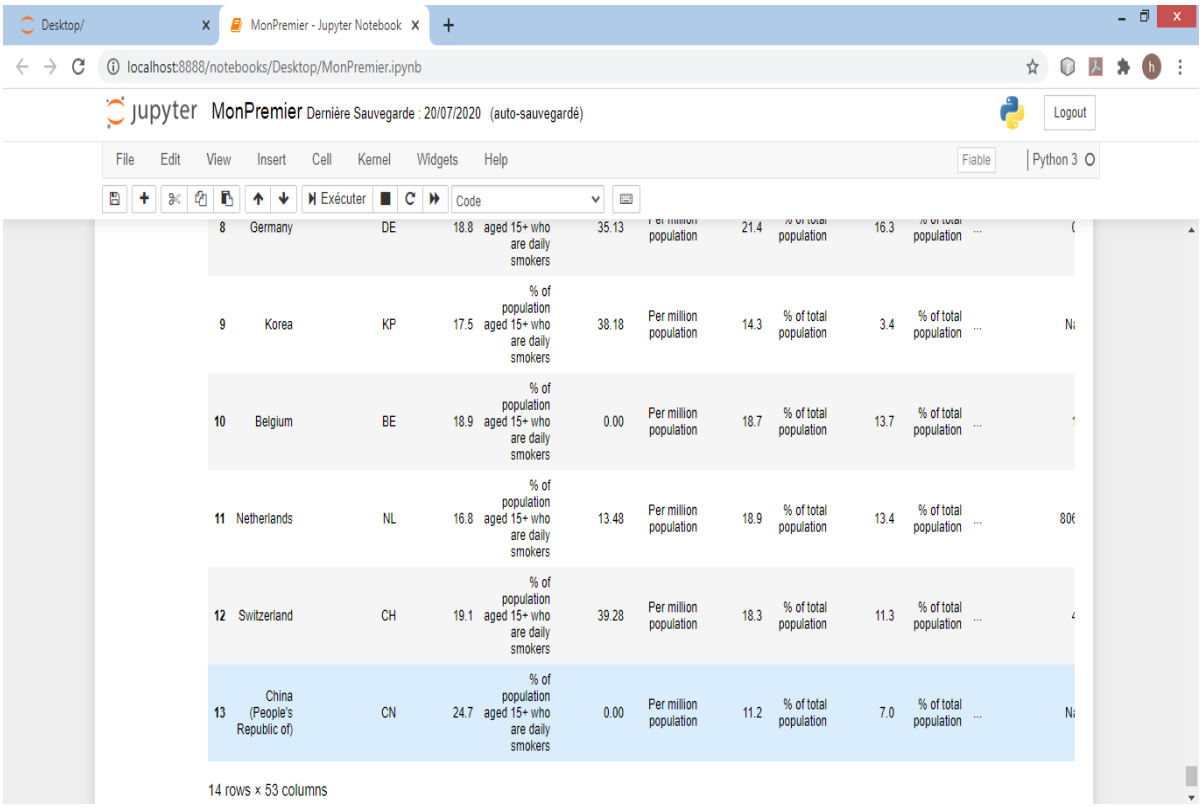


FIGURE 6.5 – Données après le nettoyage.

La **figure 6.6** représente le traitement des données dupliquées, grâce à la fonction "duplicated" qui renvoie une valeur bouléenne indiquant si la ligne est dupliquée, et la fonction "sum" qui renvoie le nombre de lignes dupliquées :

```
Entrée [15]: df.duplicated()

Out[15]: 0     False
         1     False
         2     False
         3     False
         4     False
         5     False
         6     False
         7     False
         8     False
         9     False
         10    False
         11    False
         12    False
         13    False
         dtype: bool

Entrée [16]: df.duplicated().sum()

Out[16]: 0
```

FIGURE 6.6 – Traitement des données dupliquées.

La **figure 6.7** représente le job Telend qui est responsable de chargement des données vers une base de données MongoDB :

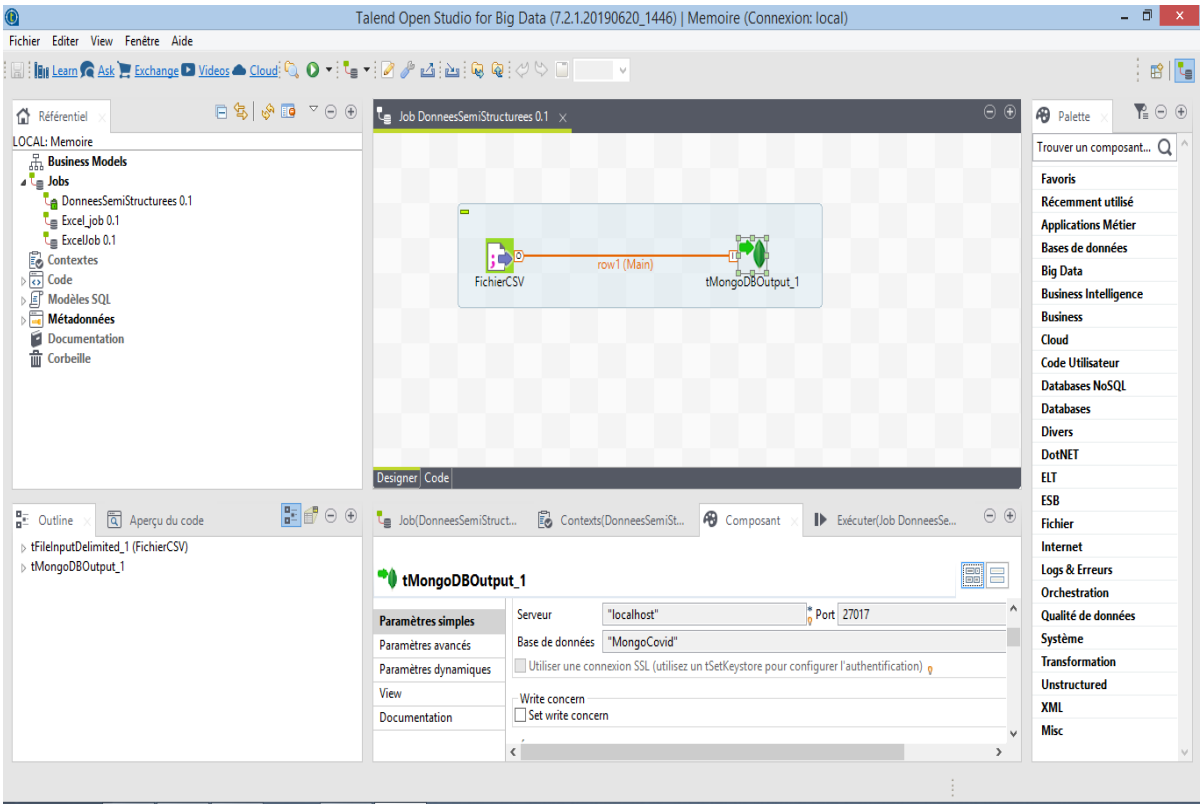


FIGURE 6.7 – job Talend.

La **figure 6.8** représente la collection générée par le job Talend sur MongoDB :

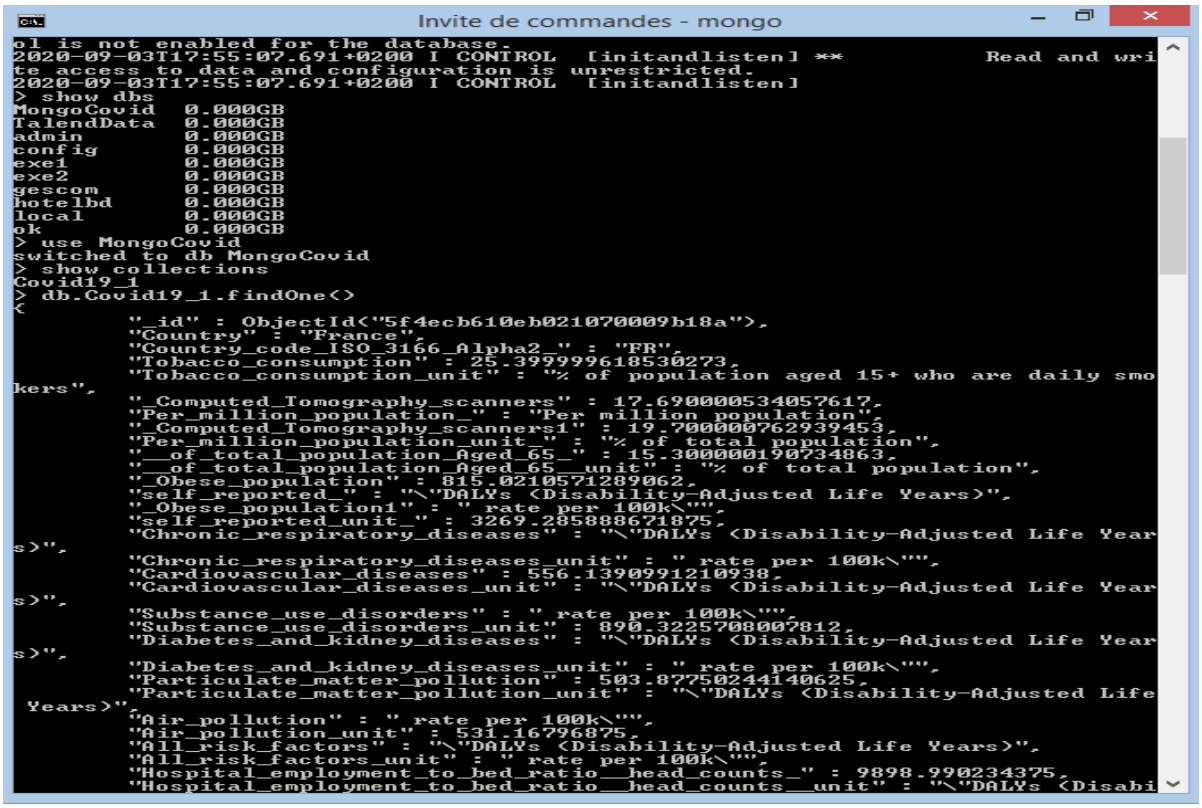


FIGURE 6.8 – Collection MongoDB.

6.7 Conclusion :

Dans ce chapitre, nous avons abordé les divers aspects liés à l'implémentation des prototypes que nous avons développés, à savoir, les technologies, les logiciels choisis en utilisant differentes sources de données pour l'implémentation de notre approche.

Chapitre 7

Conclusion générale et perspectives

Ce travail a été réalisé dans le cadre de notre projet de fin de cycle Master en informatique. Il a consisté en une amélioration d'une approche d'intégration des données dans un contexte Big Data. En effet, cette approche permet de collecter des données hétérogénes, les nettoyer, intégrer chaque source de données dans une base de données NoSQL, pour ensuite créer des ontologies locales que nous fusionnons en une seule ontologie globale sans pour autant avoir le problème de sémantique de données grâce au concept des ontologies.

Dans le premier chapitre, nous avons défini notre contexte et problématique ainsi que nos objectifs et nos contributions, nous avons également détaillé notre méthodologie de travail.

Dans le deuxième chapitre, nous avons présenté l'ensemble des notions fondamentales des Big Data, notamment l'origine, les caractéristiques, les bases de données NoSQL, sans oublier les défis et l'intégration des données massives qui sont les éléments phares de notre recherche.

Dans le troisième chapitre, nous avons présenté l'ensemble des concepts des ontologies, notamment l'origine, les composants, les langages de description, sans oublier le rôle de ces ontologies dans notre projet de recherche et enfin, leur cycle de vie.

Dans le quatrième chapitre, nous avons élaboré l'état de l'art qui représente tous les travaux connexes que nous avons synthétisés, nous avons présenté ceci dans un tableau

qui contient les grandes lignes de chaque document synthétisé, tout en suivant chaque travail par un bref paragraphe qui le résume, par la suite nous avons procédé à une analyse comparative entre les approches des documents connexes et notre approche.

Dans le cinquième chapitre, nous avons présenté en détail notre approche d'intégration de données dans un contexte Big Data utilisant les ontologies comme modèle sémantique.

Dans le sixième chapitre, nous avons abordé les divers aspects liés à l'implémentation des prototypes que nous avons développés, à savoir, les technologies, les logiciels choisis en utilisant differentes sources de données pour l'implémentation de notre approche.

Nous avons poussé le projet aussi loin que possible, il reste cependant, de nombreuses étapes à ajouter. Notamment l'implémentation de quelques étapes de notre approche, à savoir, la construction des ontologies locales à partir de la base de données chaque source de données, pour ensuite, établir l'ontologie globale que nous venons enrechir avec une ontolgie existante plus large, pour l'interroger à tarvers des reqêtes SPARQL. Nous pensons donc à enrechir notre approche et l'implémenter dans de meilleures conditions matérielles et logicielles.

La réalisation de ce projet a été riche d'enseignements sous plusieurs aspects. Elle nous a permis d'acquérir de nouvelles compétences, et de mettre en pratique les connaissances théoriques que nous avons acquis le long de notre cursus universitaire. Nous avons progressé dans de nombreux domaines notamment dans la programmation en Python, l'analyse de données massives, l'élaboration de l'état de l'art et la manipulation des base de données NoSQL.

Bibliographie

[1] Abdesalam AMRANE. Rapport sur le big data. jul 2015.

[2] Hanen ABBES, Soumaya BOUKETTAYA, and Faiez GARGOURI. Learning ontology from big data through mongodb database. *IEEE/ACS 12th International Conference of Computer Systems and Applications (AICCSA)*, pages 1–7, 2015.

[3] Bernard ESPINASSE and Patrice BELLOT. Introduction aux big data opportunités, stockage et analyse des méga données, feb 2017.

[4] Maxime VIGIER. *Les big data : une mine d'informations pour les entreprises, mémoire professionnel.* l'Université d'Evry Val d'Essonne, 2014.

[5] Loïc BREMME. le big data, https : //www.lebigdata.fr/definition-big-data, 2018. consulté le 20/10/2019.

[6] Houcine MATALLAH. *Vers un nouveau modèle de stockage et d'accès aux données dans les Big Data et les Cloud Computing.* PhD thesis, Universite Abou-bekr Belkaid, Tlemcen, 2018.

[7] Muse DAN. Structured data, https ://www.datamation.com/big-data/structured-data.html, 2017. consulté le 15/12/2019.

[8] Diego Sevilla RUIZ, Severino Feliciano MORALES, and Jesús García MOLINA. Inferring versioned schemas from nosql databases and its applications. *Springer/International Conference on Conceptual Modeling*, pages 467–480, 2015.

[9] Guy CHESNOT. *cloud computing, Big Data, parallélisme, Hadoop : stockage de données du futur.* Vuibert, 2012.

[10] Veronika ABRAMOVA and Jorge BERNARDINO. Nosql databases : Mongodb vs cassandra. *Proceedings of the international C* conference on computer science and software engineering*, pages 14–22, 2013.

[11] Ameya NAYAK, Anil PORIYA, and Dikshay POOJARY. Type of nosql databases and its comparison with relational databases. *International Journal of Applied Information Systems*, 5(4) :16–19, 2013.

[12] Patrick ZIEGLER and Klaus R DITTRICH. Data integration—problems, approaches, and perspectives. *Springer/Conceptual modelling in information systems engineering*, pages 39–58, 2007.

[13] Jens BLEIHOLDER and Felix NAUMANN. Data fusion. *ACM computing surveys (CSUR)*, 41(1) :1–41, 2009.

[14] Anirudh KADADI, Rajeev AGRAWAL, Christopher NYAMFUL, and al. Challenges of data integration and interoperability in big data. *IEEE/international conference on big data (big data)*, pages 38–40, 2014.

[15] Ines OSMAN. *Proposition d'une nouvelle méthode pour l'intégration sémantique des ontologies OWL en utilisant des alignements, Mémoire de Master*. Université de El Manar, Tunis, 2018.

[16] Robert NECHES, Richard E FIKES, Tim FININ, and al. Enabling technology for knowledge sharing. *AI magazine*, 12(3) :36–36, 1991.

[17] Rudi STUDER, V. Richard BENJAMINS, and Dieter FENSEL. Knowledge engineering : principles and methods. *ELSEVIER/Data knowledge engineering*, 25(1-2) :161–197, 1998.

[18] Oscar CORCHO, Asunción GÓMEZ-PÉREZ, and Dieter FENSEL. A roadmap to ontology specification languages. *Springer/International Conference on Knowledge Engineering and Knowledge Management*, pages 80–96, 2000.

[19] Sivadi BALAKRISHNA, M. THIRUMARAN, and Vijender Kumar SOLANKI. Iot sensor data integration in healthcare using semantics and machine learning approaches. *Springer/A Handbook of Internet of Things in Biomedical and Cyber Physical System*, pages 275–300, 2020.

[20] Nabila CHERGUI. *Une approche de mapping pour l'intégration des ontologies, Mémoire de Magister*. Université Mentouri, Constantine, 2008.

[21] Zied SELLAMI. *Gestion dynamique d'ontologies à partir de textes par systèmes multi-agents adaptatifs*. PhD thesis, Université Toulouse 3 Paul Sabatier (UT3 Paul Sabatier), Toulouse, 2012.

[22] Olivier CURÉ, Myriam LAMOLLE, and Chan Le DUC. Ontology based data integration over document and column family oriented nosql. *arXiv preprint arXiv : 1307.2603*, 2013.

[23] Sanjay AJANI. An ontology and semantic metadata based semantic search technique for census domain in a big data context. *International Journal of Engineering Research and Technology (IJERT)*, 3(2) :1–5, 2014.

[24] Václav JIRKOVSKÝ and Marek OBITKO. Semantic heterogeneity reduction for big data in industrial automation. *ITAT*, 1214, 2014.

[25] Srividya K. BANSAL and Sebastian KAGEMANN. Integrating big data : A semantic extract-transform-load framework. *IEEE*, 48(3) :42–50, 2015.

[26] KNOBLOCK Craig A. and SZEKELY Pedro. Exploiting semantics for big data integration. *Ai Magazine*, 36(1) :25–38, 2015.

[27] Richard M. KELLER, Shubha RANJAN, Mei Y. WEI, and al. Semantic representation and scale-up of integrated air traffic management data. *Proceedings of the International Workshop on Semantic Big Data*, pages 1–6, 2016.

[28] Yu FANG, Zhong JIAMING, Liu YAOHUI, and al. Semantic description and link construction of smart tourism linked data based on big data. *IEEE/International Conference on Cloud Computing and Big Data Analysis (ICCCBDA)*, pages 32–36, 2016.

[29] Hanen ABBES and Faiez GARGOURI. Mongodb-based modular ontology building for big data integration. *Journal on Data Semantics*, 1(7) :1–27, 2017.

[30] Ágnes VATHY-FOGARASSY and Tamás HUGYÁK. Uniform data access platform for sql and nosql database systems. *Information Systems*, 69 :93–105, 2017.

[31] Hanen ABBES and Faiez GARGOURI. Mongodb-based modular ontology building for big data integration. *Journal on Data Semantics*, 7(1) :1–27, 2018.

[32] Francesco GUERRA, Paolo SOTTOVIA, Matteo PAGANELLI, and al. Big data integration of heterogeneous data sources : the re-search alps case study. *IEEE/International Congress on Big Data (BigDataCongress)*, pages 106–110, 2019.

[33] Giuseppe FUSCO and Lerin AVERSANO. An approach for semantic integration of heterogeneous data sources. *PeerJ Computer Science*, 6 :e254, 2020.

[34] Xia TIM. Top covid19 countries and health demographic trend, https ://www.kaggle.com/timxia/top-covid19-countries-and-health-demographic-trend, 2020. consulté le 08/05/2020.

[35] I PIPER, G CITERIO, I HAMBERS, and al. The brainit group : concept and core dataset definition. *Acta neurochirurgica,*, 145(8) :615–629, 2003.

[36] Benjamin Marlé and Alexis PERRIER. Initiez-vous à python pour l'analyse de données, https ://openclassrooms.com/fr/courses/6204541-initiez-vous-a-python-pour-lanalyse-de-donnees/6204548-installez-python-et-anaconda, 2020. consulté le 10/05/2020.

[37] What is mongodb ? introduction, architecture, features example, https ://www.guru99.com/what-is-mongodb.html, 2020. consulté le 10/05/2020.

[38] What is nosql ?, https ://www.mongodb.com/nosql-explained, 2020. consulté le 15/05/2020.

[39] Dalila AIDENE. Talend, http ://www.open-source-guide.com/Solutions/Developpement-et-couches-intermediaires/Etl/Talend, 2016. consulté le 10/05/2020.

[40] Protégé, https ://protege.stanford.edu/products.php, 2020. consulté le 15/05/2020.

[41] Bastien L. Python : tout savoir sur le principal langage big data et machine learning, https ://www.lebigdata.fr/python-langage-definition, 2019. consulté le 15/05/2020.

[42] Bastien L. Json : tout savoir sur le format de données javascript object notation, https ://www.lebigdata.fr/json-definition, 2018. consulté le 15/05/2020.

[43] Leonidas FEGARAS. A query processing framework for large-scale scientific data analysis. *Springer/Transactions on Large-Scale Data-and Knowledge-Centered Systems XXXVIII*, pages 119–145, 2018.

[44] Bastien L. What is pandas in python ?, https ://www.educative.io/edpresso/what-is-pandas-in-python, 2020. consulté le 25/05/2020.

[45] Bibliothèques numériques de base, https ://informatique-python.readthedocs.io/fr/latest/Cours/science.htmlnumpy, 2019. consulté le 25/05/2020.