

UNIVERSITÉ LUMIÈRE LYON 2 - ICOM

PROJET DE BIG DATA MANAGEMENT

Mise en oeuvre d'un mini-lac de données textuelles

Etudiants :

Mouhamadou Mansour LO

Mame Libasse MBOUP

Encadrant :

Mr Jerome DARMONT

6 avril 2022

1 Introduction

Dans le cadre du Master 2 Data-Mining , Nous étions amené a réaliser un projet de Big Data Management portant sur la mise en place d'un mini lac de données.

Un **lac de donnée** peut se définir comme étant un système évolutif de stockage et d'analyse de données de tous types, dans leur format natif, utilisé principalement par des spécialistes des données pour l'extraction de connaissances.(Scholly et al.)

Une des caractéristiques fondamentales d'un lac de données est l'existence et la mise en disposition d'un catalogue de **métadonnées** : des métadonnées intra-objets , des métadonnées inter-objets et des métadonnées globales , permettant de comprendre la donnée, le lineage ou le versionning de la donnée , ses diverses transformations etc..

La **création** de ce lac de donnée devait nécessiter de récolter et de stocker des données dans un premier temps. Ensuite de créer , stocker puis d'instancier des métadonnées dans un second temps. Et enfin de transformer et nettoyer les données pour les besoins de l'analyse.

Nous allons présenté dans les chapitres qui suivent les diverses **démarches** et tests que nous avons suivi tout au long de ce projet , puis nous allons exposé les nombreux problèmes et difficultés rencontrés au fur et a mesure de nos avancés.

2 Présentation des données

L'idée consiste à choisir un **corpus** de textes assez varié. C'est-à-dire que ce dernier doit être un jeu de données textuelles associées à des métadonnées structurées ou semi-structurées.

Il faudra à l'issue de **pré-traitements** appliqués aux données brutes, les intégrer éventuellement dans le lac de données.








Une première phase de **stockage** a été nécessaire, ou il s'agissait de déposer le dossier récolté dans un système de fichiers d'un ordinateur.

Cet ensemble de données a été trouvé suite à de nombreux recherches sur internet . Nous avons jugé pertinent cette source car elle est particulièrement composée de **données mixtes** aux différents formats.

Le jeu de données **Feeding America** : The Historic American Cookbook contient le texte transcrit et encodé de 76 livres de cuisine américains influents conservés dans

les collections spéciales des bibliothèques de la Michigan State University(MSU). Les caractéristiques encodées dans le texte comprennent, entre autres, les recettes, les types de recettes, les instruments de cuisine et les ingrédients. Les **76 textes** ont été choisis parmi plus de 7000 livres de cuisine conservés par les bibliothèques de la MSU, car ils sont représentatifs des périodes et des thèmes de l'histoire des livres de cuisine américains, de la fin du 18e siècle au début du 20e siècle.

Index of /dinfo/feedingamerica

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
 Parent Directory		-	
 cookbook_dtd.txt	2014-11-11 09:33	21K	
 cookbook_records.xml	2014-11-12 14:58	293K	
 cookbook_text.zip	2014-11-13 17:16	17M	
 cookbook_textencoded.zip	2014-11-12 15:33	17M	
 cookbook_titlelist.txt	2014-11-14 12:16	3.5K	
 readme.txt	2016-07-14 16:28	2.4K	

Apache/2.4.41 (Ubuntu) Server at archive.lib.msu.edu Port 443

FIGURE 1 – Source de données

Source : <https://archive.lib.msu.edu/dinfo/feedingamerica/>

Le package est composé des éléments suivants :

- **cookbook_dtd** : une description de la structure des données et d'éléments bibliographique.
- **cookbook_records** : un fichier Json qui retrace les activités d'extraction et de numérisation des différents livres.
- **readme** : qui contient des détails sur la source de données
- **cookbook_titlelist** : qui contient les titres et ids des diverses livres de cuisines.
- **cookbook_txt** : qui contient les livres de cuisine parsés
- **cookbook_xml** : qui contient les livres de cuisines sous format XML.

Exemple de recette tirée d'un livre : BAKED BEEF HEAD. (Without cooking utensils.)

Dig in the ground a hole of sufficient size and build a fire in it. After the fuel has burned to coals put in the head, neck downward. Cover it with green grass, coals, and earth. Build a good fire over the buried head and keep it burning for about six hours.

Unearth the head and remove the skin. A head treated in this way at night will be found cooked in the morning. The head of any animal may be cooked in this way.

Ingrédients : ['BEEF HEAD.', 'coals', 'head,', 'neck', 'coals,', 'head', 'head', 'skin.', 'head', 'head']

3 Travail réalisé

La **première étape** a été de prendre en main l'ensemble des fichiers du dossier. Nous avons par la suite chargé les fichiers sur Python afin de réaliser quelques **pré-traitements** la-dessus.

En autres comme pré-traitement nous avons parsé tous les livres qui étaient sous format **XML** pour tirer grâce aux diverses **balises**, les **recettes** des livres, les **ingrédients** des livres, les **ustensiles**, l'auteur des livres, la date de publication des livres, les caractéristiques des livres : type de **nourriture**, **origine ethnique** nourriture, zone **géographique** ...

Ces premiers pré-traitement ont permis d'obtenir des métadonnées de type **inter-objet** permettant de regrouper les livres et recettes par thématique (Voir fichier projet `Parse_XML.ipynb`)

Concernant les métadonnées **intra-objet** nous avons essayé de prime abord, d'utiliser **ApacheTika** pour extraire des métadonnées à partir des fichiers **brutes**. Cette méthode ne s'est pas montrée très efficace pour fournir les propriétés comme **nom du fichier**, **taille**, **emplacement**, **date de création** et **dernière modification**...). Raison pour laquelle on a opté pour coder quelques scripts sur Python qui permettront de répondre à cette question. (Voir fichier projet `_data_lake.ipynb`)

Avec la librairie **OS**, nous sommes parvenus à extraire toutes ces propriétés et finalement d'avoir un fichier Json pour les 76 films associés à leurs propriétés. Il s'agit là des informations qu'on peut avoir pour chaque fichier d'un répertoire en faisant un clic-droit sur celui-ci.

A la suite, pour stocker les **métadonnées** nous avons fait appel à **mongoDB** avec le couple (mongoDB Server et pymongo sur Python).Ceci a été possible en créant une base de données métadonnées avec premièrement une collection métadonnées_inter,

suivie d'une seconde collection metadonnées_intra. Les détails de notre démarche sont présentes dans le fichier **MongoDB.ipynb**. Et les images qui suivent montrent l'interface **mongoDB** en local.

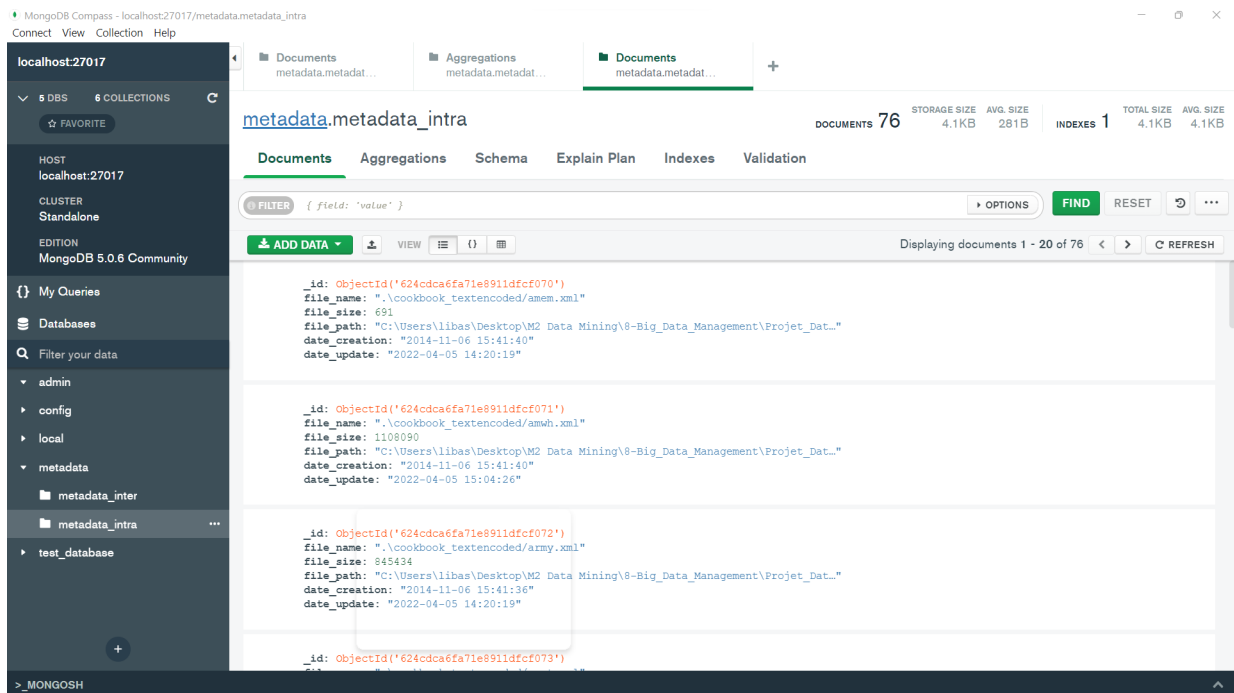


FIGURE 2 – Sur MongoDB : metadonnée intra-objet

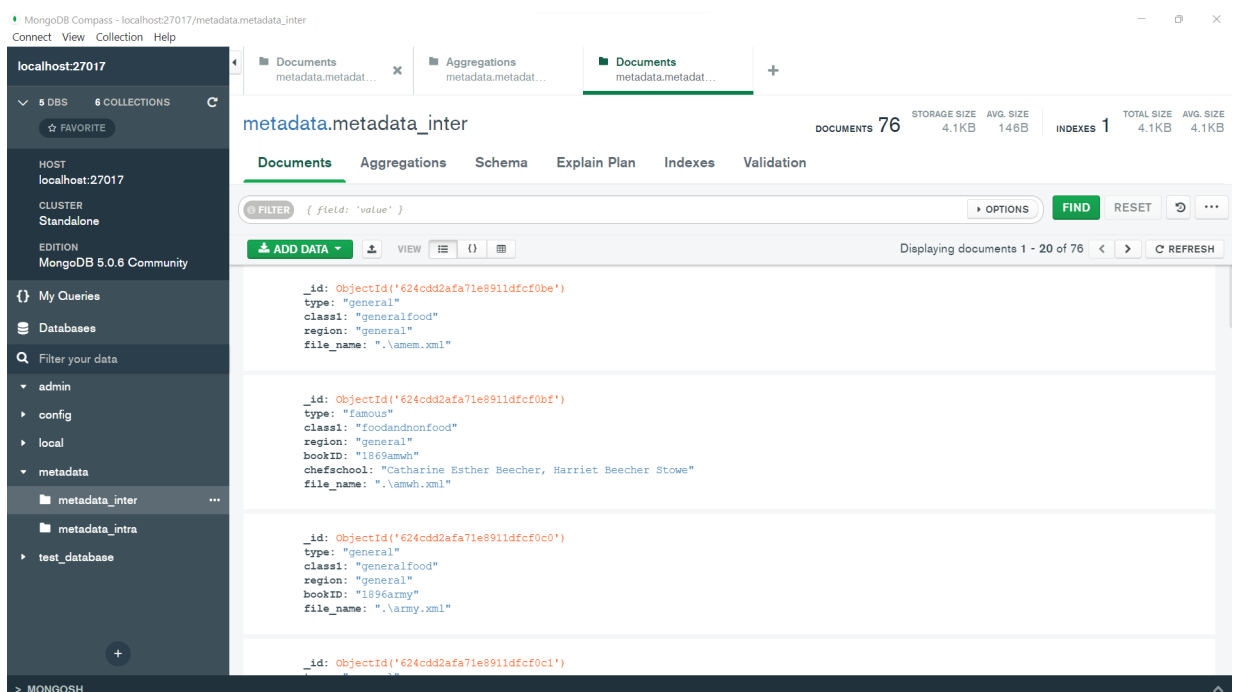


FIGURE 3 – Sur MongoDB : metadonnée inter-objet

4 Problèmes rencontrés - Perspectives

- **ApacheTika** : Avec cet outil, on a pas eu des métadonnées intéressantes même si Apache Tika reste une technologie assez robuste à ce sujet. Nous avons pensé au fait que le format des fichiers pouvait ne pas coïncider pour extraire les informations adéquates.
- **ElasticSearch - Kibana** : Nous avons eu à rencontrer énormément de problèmes lors de l'utilisation de la suite d'**Elastic Search** et de **Kibana** pour la visualisation.

De l'installation à l'utilisation, nous n'avons malheureusement pas pu utiliser ces différents outils pour générer un **index inversé** des documents ou pour faire de la **visualisation** et **BI** sur notre corpus.

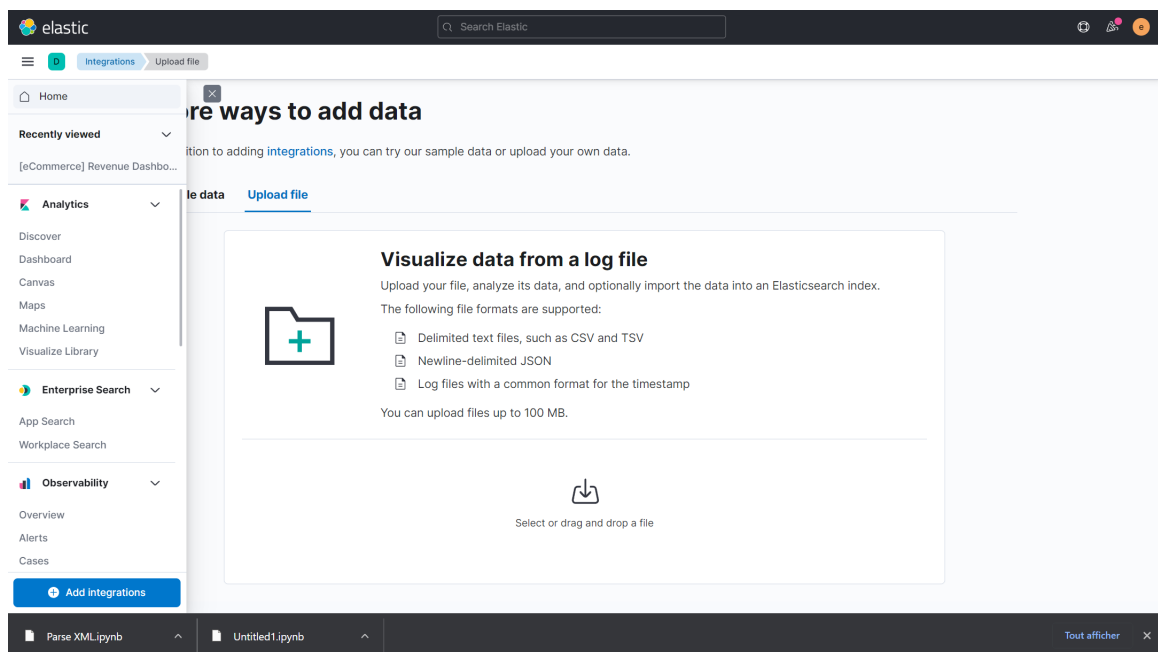


FIGURE 4 – Instanciation de notre environnement Kibana

5 Conclusion

Ces travaux nous ont permis de consolider nos connaissances sur les technologies Big Data notamment celles liées à la programmation Python, en nous familiarisant avec de nombreuses bibliothèques de Data Science (**MongoDB**, **ApacheTika**, **PyMongo**, **Os**, **BeautifulSoup**, etc.). Nous avons eu l'occasion de mettre en pratique les différentes techniques et connaissances acquises durant le cours, notamment, les théories sur les lacs de données et métadonnées. Hormis les connaissances acquises sur les notions relatives aux **lacs de données**, ce travail nous a permis aussi de collaborer en équipe qui est un aspect très

important dans notre formation éventuellement pour un futur proche, et d'améliorer nos capacités d'analyse. La principale difficulté que nous avons rencontrée dans ce projet a été la démarche générale à adopter pour inclure un **index inversé** des termes des documents textuels concernant les métadonnées **globales**.