UFR
DE MATHÉMATIQUES
ET INFORMATIQUE

Université de Paris

Projet BI:

Business Intelligence

EXTRACTION DES CONNAISSANCES À PARTIR DU TEXTE

Professeur :

Rafika Boutalbi

Étudiant :

Mohamed Abdelhadi Boudjemai

Année universitaire : 2020 – 2021

Table des matières

1. Introduction:

Business Intelligence est le processus d'analyse de données dirigé par la technologie dans le but de déceler des informations utilisables ou importantes. Elle regroupe une large variété d'outils, d'applications et de méthodologies permettant de collecter des données en provenance de systèmes internes et de sources externes, de les préparer pour l'analyse, de les développer et de lancer des requêtes au sein de ces ensembles de données.

La plateforme d'intégration des données Qlik concrétise les Data en transformant les données brutes en données exploitables, fiables, actualisées, faciles à trouver et immédiatement disponibles dans Qlik Sense.

Dans ce projet, on dispose d'un fichier texte contenant des informations sur des articles scientifiques parus dans des revues et conférences comme le titre de l'article, le ou les auteurs, l'année de publication, nom de la revue, etc… On va ainsi traiter ces données en créant des dataframes contenant les informations sur des articles (titres, résumés, les identifiants des auteurs et leur articles) sur une application Qlik Sense et en utilisant la langage python.

2. Objectif :

Dans ce projet, on dispose d'un fichier texte « DBLP_new_version » contenant des informations sur des articles scientifiques parus dans des revues et conférences. Ces informations comprennent :

- Le titre de l'article,
- Le ou les auteurs,
- L'année de publication,
- Nom de la revue (ou de la conférence)
- Les citations entre articles.

Nous souhaitons faire une étude sur ce document, tout en passant par une série d'opérations en utilisant la langage python et l'outil Qlik Sense.

Sur cette étude on va passer par plusieurs étapes (figure 1), tout d'abord on fait un traitement de données (traitement de texte) afin d'extraire les informations séparément sous forme d'un dataframe. La deuxième étape est le nettoyage du texte de ce dataframe, afin d'appliquer les fonctions nécessaires pour obtenir les matrices doc termes, ensuite on appliquera un clustering à l'aide de l'algorithme de Louvain après avoir construit les graphes co-termes. Ainsi, réaliser un consensus entre les partitions découvertes par les clustering précédents, et le tout sera résumé dans un tableau de bord de l'outil Qlik Sense contenant une analyse descriptive détaillée de la base de données et les résultats du clustering.

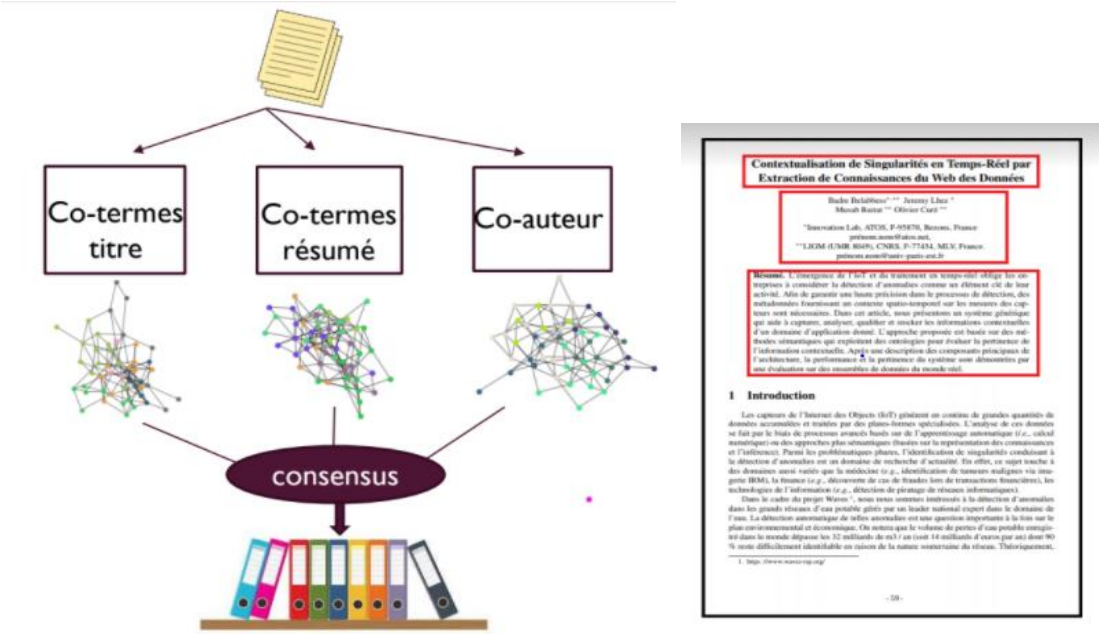


Figure 1 Informations sur des articles scientifiques

Il faut savoir que plusieurs calculs ont été évité, à cause du volume très grand de notre ficher texte et les fichiers CSV obtenu (comme par exemple la matrice co-Auteur, et citations), le chargement et le traitement des données prend beaucoup de temps. De plus, Qlik Sense limite la taille des fichiers importés.
On a essayé d'exploiter au maximum notre dataframe pour avoir des informations pertinentes.

3. Traitements de données (traitement de texte) :

Notre jeu de données est un fichier texte (figure 2), il est structuré de la manière suivante chaque ligne est représentée par un caractère spéciale : les titres par #*, les auteurs #@ ainsi de suite voir la figure 3.

```
#*Improved Channel Routing by Via Minimization and Shifting.
#@Chung-Kuan Cheng
David N. Deutsch
#t1988
#cDAC
#index131751
#%133716
#%133521
#%134343
#!Channel routing area improvement by means of via minimization and via shifting in two dimensions (compac
can be minimized by wire straightening. The implementation of algorithms for each of these procedures has
Example
 the standard channel routing benchmark
 that is more than 5% smaller than the best result published heretofore. Suggestions for possible future w

#*A fast simultaneous input vector generation and gate replacement algorithm for leakage power reduction.
#@Lei Cheng
Liang Deng
Deming Chen
Martin D. F. Wong
```

Figure 2 Jeu de données en fichier texte

Chaque information est introduite par un caractère spécifique voir la figure 3

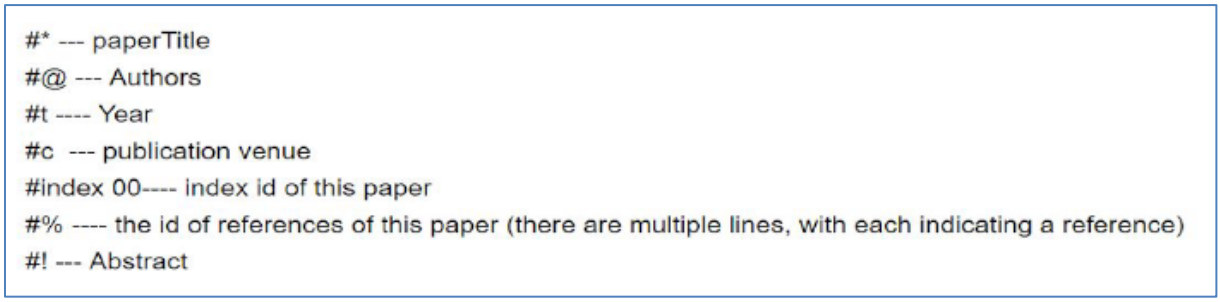


Figure 3 introduction de chaque information par un caractère spécifique

Après le Chargement du fichier texte sous python et le traitement de données *grâce à la fonction remplissage, qui traite ligne par ligne notre document on se basant sur les* caractères spécifiques *pour remplir notre data frame figure 4.*

L'algorithme ou la fonction du traitement est bien clair et expliqué sur le code python.

	Venue	Year	authers	Titre	NbrAuther	id	ListCitation	NbrCitation	Abstract
0	DAC	1988	Chung-Kuan Cheng,vid N. Deutsch	Improved Channel Routing by Via Minimization a...	2	131751	133716,133521,134343	3	Channel routing area improvement by means of ...
1	DAC	2006	Lei Cheng,ang Deng,ming Chen,rtin D. F. Wong	A fast simultaneous input vector generation an...	4	131752	132550,530568,436486,134259,283007,134422,2821...	8	Input vector control (IVC) technique is based...
2	DAC	1992	Kwang-Ting Cheng,-Keung Tony Ma	On the Over-Specification Problem in Sequentia...	2	131756	455537,1078626,131745	3	The authors show that some ATPG (automatic te...
3	DAC	2005	Lerong Cheng,oebe Wong,i Li,n Lin,i He	Device and architecture co-optimization for FP...	5	131759	214244,215701,214503,282575,214411,214505,132929	7	Device optimization considering supply voltag...
4	DAC	1989	Wu-Tung Cheng,ng-Lin Yu	Differential Fault Simulation - a Fast Method ...	2	131760	131744,806030	2	A new fast fault simulator called

Figure 4 Exemple de data frame

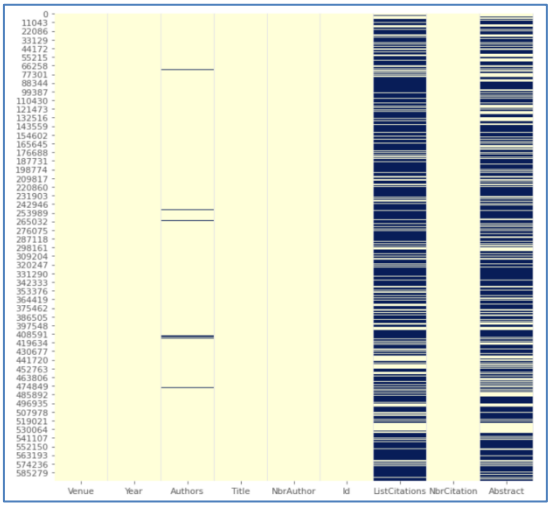
Les trois figures ci-dessous (figure 5: a,b ,c) nous donnent des informations sur la taille, nom de colonnes, types et les données manquantes sur notre dataframe.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 596277 entries, 0 to 596276
Data columns (total 9 columns):
 #   Column         Non-Null Count   Dtype
---  ------         --------------   -----
 0   Venue          596276 non-null  object
 1   Year           596277 non-null  int64
 2   Authors        591883 non-null  object
 3   Title          596277 non-null  object
 4   NbrAuthor      596277 non-null  int64
 5   Id             596277 non-null  int64
 6   ListCitations  149743 non-null  object
 7   NbrCitation    596277 non-null  int64
 8   Abstract       198887 non-null  object
dtypes: int64(4), object(5)
memory usage: 40.9+ MB
```

(a)

	Total	Percent
ListCitations	446534	74.89
Abstract	397390	66.65
Authors	4394	0.74
Venue	1	0.00
NbrCitation	0	0.00
Id	0	0.00
NbrAuthor	0	0.00
Title	0	0.00
Year	0	0.00

(b)



(c)

Figure 5 Présentation des données manquantes en dataframe

On remarque qu'il y a beaucoup de données manquantes sur la colonne citations et résumé, et aussi quelques données manquantes sur la colonne auteurs (les lignes qui contient des 'NA').

4. Nettoyage et la construction des matrices Documents-termes :

Pour Construire la matrice Documents-termes des titres d'articles ainsi que pour les résumés (Abstract) on doit faire un traitement du texte a l'aide de la bibliothèque Python `nltk`, d'abord enlever tout ce qui est 'NA', ensuite la suppression des caractères spéciaux, la suppression des chiffres et les stops words, puis une étape de lemmatisation (remettre le mot a sa racine) des colonnes Abstarct et titre.

Une fois ces transformations faites on va pouvoir utiliser le `fit_transform` on le construit à partir du `CountVectorizer du package sklearn` cette fonction nous permet de transformer notre donnée en un tableaux documents-mots (voir le code python).

Pour construire la matrice Documents-auteurs, on crée une liste des auteurs uniques après on rend le texte en minuscule puis on crée une matrice vide (plein de zéro) taille en ligne des documents et en colonne la liste des auteurs uniques ensuite en fait un parcours pour mettre un 1 si l'auteur écrit l'article (voir le code python).

5. Analyse descriptive des data par Qlik Sense:

Une fois ces quatre 'dataframe' ont été créés :

- Un 'dataframe' contenant les informations articles.
- Un 'dataframe' contenant la matrice documents-termes (titre et résumé).
- Un 'dataframe' contenant les identifiants des auteurs.
- Un 'dataframe' contenant la matrice documents-auteurs.

On génère les fichiers CSV pour ces tableaux de données et on les charge sur une application Qlik Sense.

La figure 6 présente un tableau de bord contenant une analyse descriptive détaillée de la base de données.

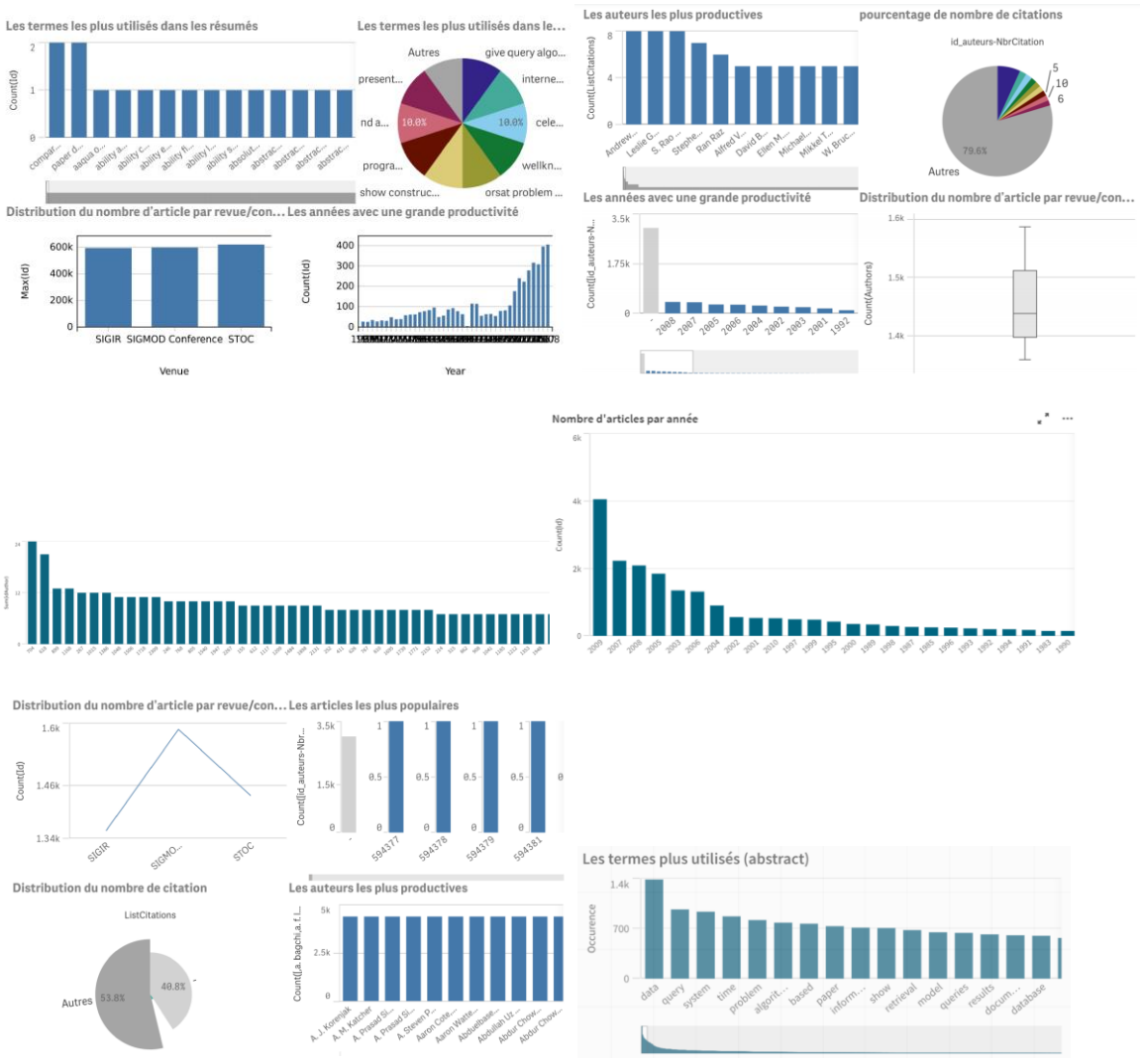
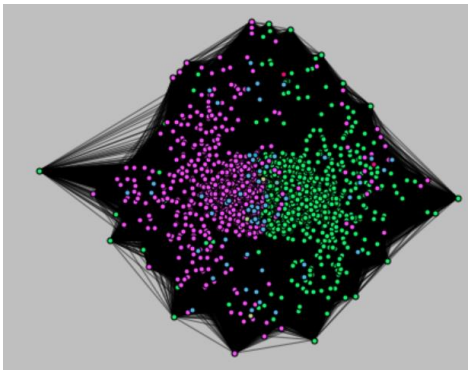


figure 6 tableau de bord contenant une analyse descriptive détaillée de la base de données

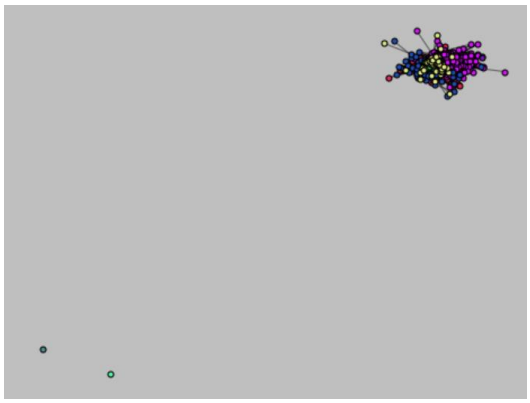
6. Les graphes co-termes :

Pour construire les quatre graphes : co-termes (titre), co-termes (abstract), co-auteurs, on calcule nos matrices d'adjacences le produit scalaire des matrices Documents-termes avec leurs transposés, puis donner ces matrices d'adjacences à networkx pour les transformer en graphe.

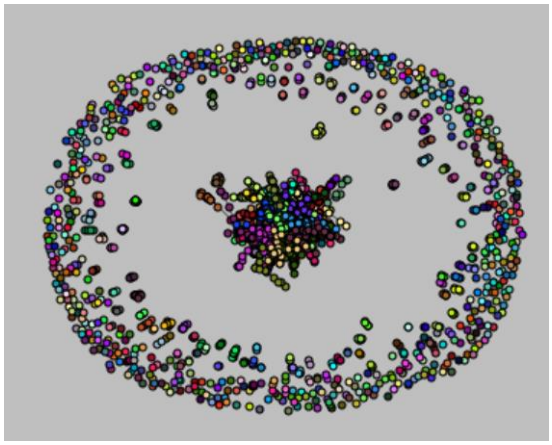
Voici ci-dessous les trois graphes co-termes figure 7 (a, b et c).



(a)



(b)



(c)

figure 7 Les trois graphes co-termes

7. Clustering (l'application de l'algorithme de Louvain) :

Après l'installation du package «community» à l'aide de la commande pipinstallpython-louvain, l'installation du package «networkx» à l'aide de la commande pipinstallnetworkx et l'installation de «metis» sudoapt-get install metis et l'installation du package «Cluster_Ensembles» avec la commande pipinstallCluster_Ensembles, on fait un clustering avec l'algorithme de Louvain de nos partitions.

La figure 8 explique l'algorithme de Louvain :

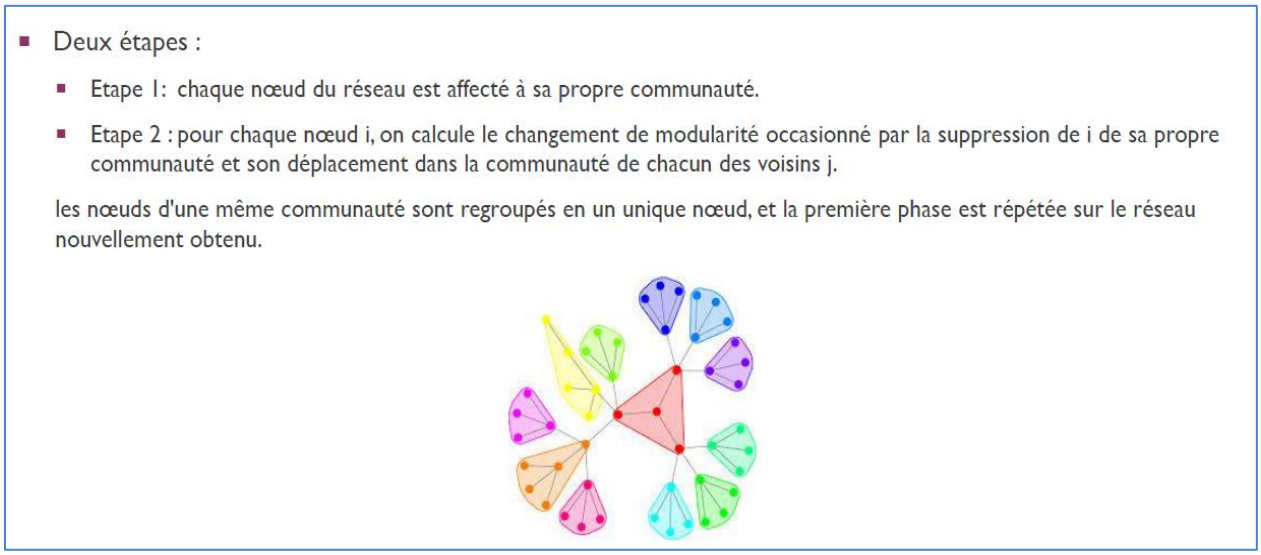


Figure 8 Principes étapes de l'algorithme de Louvain

8. Analyser des résultats du clustering (Qlik Sense) :

Voici ci-dessous figure 9 un second tableau de bord permettant d'analyser les résultats du clustering.

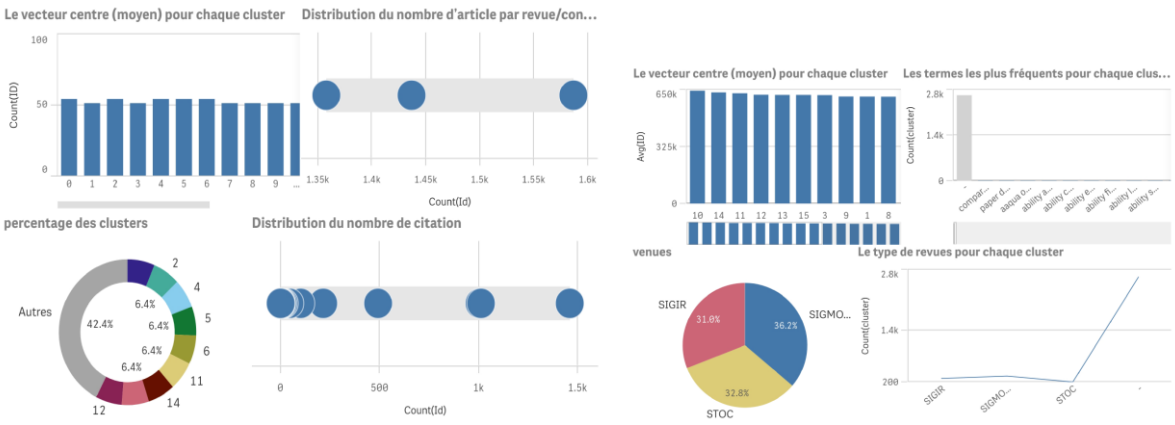
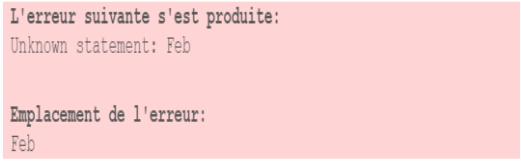


figure 9 Tableau de bord pour analyse des résultats du clustering

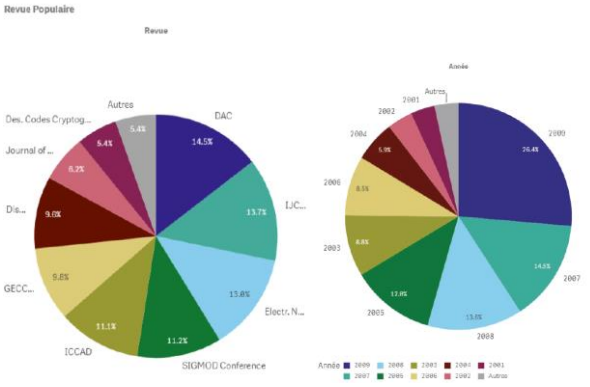
Il faut savoir que plusieurs calculs ont été évité, à cause du volume très grand de notre ficher texte et les fichiers CSV obtenu (comme par exemple la matrice co-Auteur, et citations), le chargement et le traitement de la donnée prend beaucoup de temps. De plus, Qlik Sense limite la taille des fichiers importés.



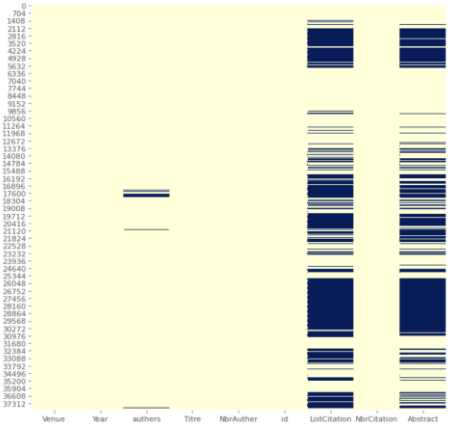
9. Bonus :

Pour cette partie j'ai refait toute l'étude sur le premier jeu de données « DBLP_Subset ».

Voir le code phyton ci-joint par mail et voici ci-dessous quelques résultats de l'analyse.




```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 37966 entries, 0 to 37965
Data columns (total 9 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   Venue         37964 non-null  object
 1   Year          37966 non-null  object
 2   authers       37549 non-null  object
 3   Titre         37966 non-null  object
 4   NbrAuther     37966 non-null  object
 5   id            37966 non-null  object
 6   ListCitation  18746 non-null  object
 7   NbrCitation   37966 non-null  object
 8   Abstract      20557 non-null  object
dtypes: object(9)
memory usage: 2.6+ MB
```



	Total	Percent
ListCitation	19220	50.62
Abstract	17409	45.85
authers	417	1.10
Venue	2	0.01
NbrCitation	0	0.00
id	0	0.00
NbrAuther	0	0.00
Titre	0	0.00
Year	0	0.00

	Venue	Year	authers	Titre	NbrAuther	id	ListCitation	NbrCitation	Abstract
count	37964	37966	37549	37966	37966	37966	18746	37966	20557
unique	283	75	32657	36923	18	37963	18350	67	20525
top	J. Symb. Log.	2009	[David B. Lomet]	Preface.	1	132139	[86]	0	This paper presents the findings of a five-ye...
freq	3402	4866	75	123	27178	3	18	19220	3

10. Conclusion :

A l'heure où les avantages compétitifs et la performance dépendent de plus en plus de la capacité à collecter, maîtriser et valoriser des données volumineuses et diverses, la Business Intelligence devient un outil clé pour toutes les organisations, des plus grandes aux plus petites.

Les programmes de business intelligence peuvent avoir de nombreux bénéfices pour les sociétés académiques et entreprises. Ils permettent d'accélérer et d'améliorer la prise de décision, d'optimiser les processus internes, d'augmenter l'efficience d'exploitation, de générer de nouveaux revenus, et de prendre l'avantage sur la concurrence.

Dans ce projet, on a été disposé d'un fichier texte contenant des informations sur des articles scientifiques parus dans des revues et conférences comme le titre de l'article, le ou les auteurs, l'année de publication, nom de la revue, ...

On a traité ces données en créant des dataframes contenant les informations articles (titres, résumés, les identifiants des auteurs et leur articles) sur une application Qlik Sense et en utilisant la langage python. Notre travail est passé par plusieurs étapes. En premier lieu, un traitement de données (traitement de texte) afin d'extraire les informations séparément sous forme d'un dataframe.

La deuxième étape etait le nettoyage du texte du dataframe , afin d'appliquer les fonctions nécessaires pour obtenir les matrices doc termes. On a ensuite appliqué un clustering à l'aide de l'algorithme de Louvain ainsi on a pu construire les graphes co-termes. Dela, on peut réaliser un consensus entre les partitions découvertes par les clustering précédents, et le tout sera résumé dans un tableau de bord de l'outil Qlik Sense contenant une analyse descriptive détaillée de la base de données et les résultats du clustering.