Université Paris Descartes
UFR de Mathématiques et Informatique
Master 1 Informatique

Rapport de projet de Big Data
Thème : Clustering de text

Réalisé par:

Yanis AIT HAMMOU
Assia BOURAI
Said SADEG
Mohamed BOUDJEMAI

Chargé de Cours/TD : Themis PALPANAS

2019-2020

1.Introduction :

En raison de la croissance explosive des données issue du web, de l'avènement des moteurs de recherche et des réseaux sociaux dans notre vie quotidienne, l'utilisation du machine learning est devenue primordial pour pouvoir analyser ces données.

Parmi toutes ces méthodes de traitement de données on peut citer le clustering qui constitue l'une des tâches les plus importantes pour le traitement des données textuelles.

Dans la littérature il existe plusieurs algorithmes de clustering notamment le K-means qui est l'un des algorithmes les plus populaire pour le clustering, la Classification Ascendante Hiérarchique (CAH), le DBSCAN (density-based spatial clustering of applications with noise) … etc. Mais aujourd'hui, de nouvelles méthodes de clustering sont apparues comme le spectral clustering ainsi que l'ensemble clustering.

L'ensemble clustering est une méthode qui permet de combiner plusieurs modèles de clustering afin de produire un modèle de meilleur qualité et qui compensera les défauts des modèles individuels

Cet algorithme se base sur deux étapes essentielles, la première consiste à générer un ensemble de partitions de base en utilisant un ou plusieurs algorithmes de clustering , ensuite vient l'étape de construction de la partition finale en utilisant une fonction consensus qui prendra en entrées les partitions de base générées par l'étape précédente pour ensuite produire une partition finale appelée aussi partition consensus.

Parmi les approches de l'ensemble clustering on trouve l ' Adaptive Evidence Accumulative Clustering (AdaEAC) [1], qui se base sur un algorithme de clustering avec pondération d'objets qui concentre le processus d'apprentissage sur les objets qui le plus de poids, ainsi que l'utilisation d'un degré de confiance pour déterminer l'affectation des objets aux clusters.

Dans ce travail nous allons implémenter une variante de l'algorithme de AdaEAC pour le traitement des données textuelles, pour cela nous allons appliquer les techniques de pré-traitements de texte standard pour notre dataset pour pouvoir ensuite utiliser les étapes fondamentales de l'algorithme AdaEAC améliorées en utilisant différentes techniques.

Ce rapport sera organisé en trois sections principale, dans un premier temps nous présenterons l'état de l'art. Dans cette section nous allons présenter une revue de la littérature concernant les différentes méthodes utilisées pour le texte clustering.

Dans la deuxième section de ce rapport nous allons détailler le fonctionnement de l'algorithme la méthode implémentée et nous expliquerons toutes les étapes.

Nous verrons ensuite la partie expérimentations; dans cette partie nous allons tout d'abord présenter les techniques de prétraitement de texte qu'on a utilisé, et ensuite nous allons exposer les résultats d'exécution de cet algorithme sur les datasets et on le comparera aux résultats des méthodes standard du clustering.

2. Travaux connexes :

Parmi les méthodes clustering de texte qui existe dans la littérature on trouve celle introduite dans [2] qui est une méthode qui consiste à utiliser une classification itérative des données aberrantes issues d'un algorithme de clustering classique et cela dans le but d'améliorer les performances de cet algorithme, cette approche consiste à utiliser les données labellisées non aberrantes issue de l'algorithme de clustering classique pour entraîner un modèle de classification supervisé et ensuite déterminer les clusters auxquels appartiennent les données aberrantes.

On trouve aussi l'algorithme DSNN introduit en [3] et qui consiste à utiliser une variante distribué de l'algorithme SNN (shared Nearest Neighbors) cet algorithme consiste à attribuer un individu au cluster avec lequel il a le plus de voisins, cet algorithme s'appuie sur une

architecture maître-esclave et le décompose en deux étapes, la première consiste à traiter les données dans chaque nœud et produire des points centraux qui sont envoyés par la suite au maître qui dans la deuxième étape utilise un SNN sur ces points, et au final un ensemble de points représentatif labélisé définissant les points centraux qui résume la collection.

On trouve aussi l'approche exposé en [4] et qui consiste à consiste à inclure des données spatiales dans le processus de clustering afin de permettre de classer des tweets dans des clusters, cet algorithme est basé sur l'approche DBSCAN qui se base sur la densité des points dans l'espace pour former des clusters, cette méthode vise à inclure des coordonnées géographiques afin d'augmenter les performances de l'algorithme DBSCAN.

On trouve aussi dans [5] une approche qui permet d'améliorer clustering des textes courts qui sont généralement plus difficile à clusteriser vue la nature des ces textes qui sont petit et qui contiennent des abréviations et des langages non formel, pour cela l'approche utilise trois étapes pour réaliser le clustering, premièrement un graphe de termes où les nœuds et les arcs sont pondérés, ensuite un regroupement des mots significatifs dans le texte est effectué en utilisant la notion de voisinage pour ensuite construire les clusters.

La méthode présentée dans l'article [6] quant à elle a pour objectif principal de montrer que les récents algorithmes de classification peuvent être améliorés en fonction du processus de pré-traitement de texte utilisé . les auteurs ont présenté une méthode de pré-traitement des vecteurs caractéristiques.

La méthode proposée en [7], se concentre beaucoup plus sur le NLP des texts dits courts. Il s'agit d'une approche qui apprend les caractéristiques discriminantes à la fois d'un auto-encodeur et d'un embedding de phrases, puis utilise les affectations d'un algorithme de clustering comme supervision pour mettre à jour les poids de l'encodeur.

Le modèle proposé suit trois étapes: dans la première phase, les textes courts sont intégrés à l'aide du SIF (Smooth Inverse Frequency) embedding. Après cela , au cours d'une phase de pré-formation, un auto-encodeur profond est appliqué pour coder et reconstruire les intégrations SIF de texte court . La dernière étape est celle d'autoformation, où les affectations de clusters sont utilisées en tant que distribution cible auxiliaire pour affiner et réviser conjointement les poids de l'encodeur.

Le travail présenté en [8] aborde un nouveau schéma de représentation de texte en regroupant les mots selon leur sémantique et en les regroupant ensemble pour obtenir un ensemble de vecteurs cluster, qui sont ensuite concaténés comme représentation textuelle finale.

En d'autres termes, les auteurs de cet article, proposent d'augmenter la représentation textuelle à partir d'un niveau supérieur: le niveau cluster. Les mots d'un un texte sont divisés en différents clusters sémantiques. Des représentations de cluster sont par la suite obtenues en combinant les intégrations contextuelles des mots ensemble en fonction de leur distribution de probabilité de cluster. Les représentations de cluster sont ensuite concaténées en tant que représentation finale du texte.

L'article [9] traite une méthode qui sert pour la synthèse de texte, en d'autres termes il traite le processus consistant à extraire les informations les plus importantes d'un texte. À l'aide d'une technique de clustering, cet article a suivi deux étapes:

La première étape consiste à découvrir les sujets du texte en regroupant les phrases à l'aide de la méthode k-means. Chaque phrase est représentée comme un vecteur où chaque composante reflète le poids d'un terme correspondant. La mesure de similitude a été calculée par la fréquence du terme dans les phrases.

La deuxième étape est la sélection de phrases saillantes à partir de grappes en optimisant une fonction objective qui permet de maximiser la

couverture des phrases, leur diversité (concision) et la longueur.

Pour résoudre le problème d'optimisation, les auteurs ont proposé un algorithme d'évolution différentielle adaptative avec une nouvelle stratégie de mutation (COSUM).

Dans l'article [10], les auteurs ont proposé une technique de modélisation de sujet qui découvre les sujets sémantiques à partir de documents biomédicaux. Pour cela, ils utilisent la fréquence des termes locaux et la fréquence des termes globaux via le modèle Bag Of Words. La technique proposée utilise la classification et le clustering pour l'exploration de texte avec une probabilité de sujets dans les documents. La classification est effectuée par le biais d'un classificateur d'analyse discriminante tandis que la classification est effectuée via la classification k-means.

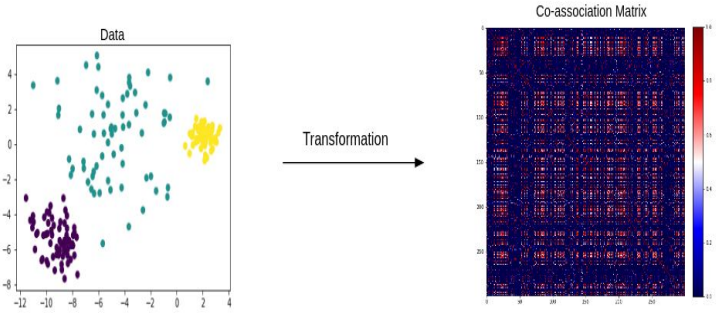

Figure. 2: Transformation d'un noyau de base vers une matrice de co-association.

3. Présentation de la méthode implémentée:

Dans la première étape de cet algorithme **figure.1,** plusieurs partitions de base sont générées en parallèle en utilisant l'algorithme KMeans, avec un nombre de clusters important, afin d'extraire les informations locales des clusters. Ensuite, ces informations seront insérées dans la matrice de co-association, où chaque valeur correspond à la fréquence d'apparition d'une paire d'objets dans le même cluster dans les partitions de bases. On peut donc voir ça comme une transformation d'un noyau de base vers une matrice de co-association, tel que le montre la **figure.2**.

La **figure 3** , montre la matrice co-association réordonnée en utilisant la méthode d'évaluation visuelle de la densité.

On peut voir ici qu'il y a trois clusters, le cluster c3 est bien distingué des autres, et on remarque qu'il y a une confusion entre c1 et c2, ce qui fait que si on applique un modèle directement sur cette matrice, on risque d'avoir de mauvais résultats, car des erreurs sont introduites lors de la phase de transformation. Ces erreurs sont dans les rectangles jaunes de la figure, il faut donc les supprimer afin d'avoir des résultats plus précis.

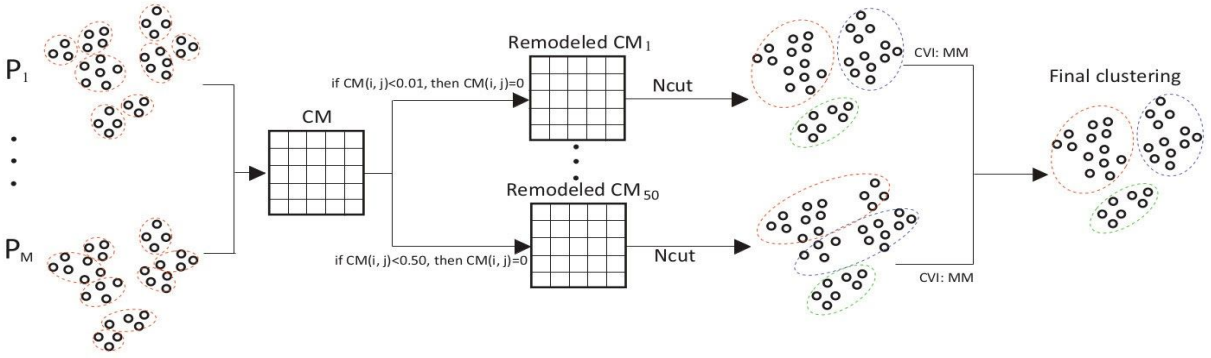

Figure. 1: Plan d'exécution de l'algorithme d'ensemble clustering

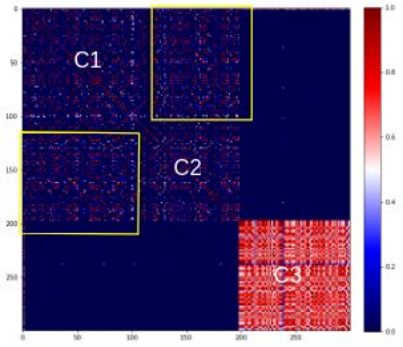

Figure. 3: La matrice de co-association réorganisée en utilisant la méthode **VAT**.

Sur la **figure.4**, nous avons sur la droite les valeurs prises par les informations positives, c'est-à-dire, les fréquences des paires d'objets qui appartiennent aux même clusters, et à gauche on voit les informations négatives c'est-à-dire les erreurs, ici on suppose que les informations négatives correspondant aux fréquences qui ont des valeurs inférieures à 0,5, ce n'est pas toujours vrai, donc, si on supprime ses erreurs directement c'est-à-dire les fréquences qui ont des valeurs en dessous de 0.5, on risque de supprimer beaucoup d'informations positives.

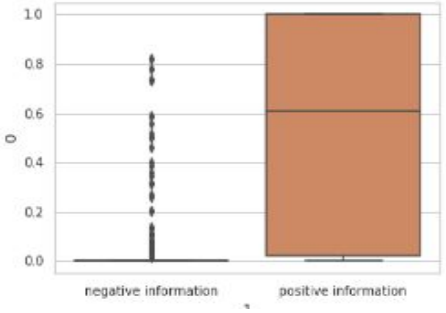

Figure. 4: boxplote correspond aux informations positives / négatives de la matrice de co-association.

Pour remédier à ce problème, dans la deuxième étape de cet algorithme, nous allons effectuer des suppressions à plusieurs niveaux, en utilisant une boucle avec un seuil qui va de 0 à 0,5 avec un pas de 0,01, à chaque itération, une matrice de co-association CMI est générée dont les fréquences qui ont des valeurs en dessous de ce seuil seront supprimés, puis l'algorithme de segmentation d'image est appliqué à chaque matrice générée CMI. Comme le montre la **figure.5**, la matrice sera transformée en graphe pondéré, puis ce graphe sera divisé en plusieurs composantes connexes.à la fin de cette étape, nous aurons exactement 50 partitions candidates finales. Dans la dernière étape de cet algorithme, une seule partition sera sélectionnée à l'aide d'une métrique d'évaluation interne qui utilise uniquement les informations de la matrice de co-association, la méthode que nous avons utilisée ici est basée sur le degré de confiance d'appartenance d'un objet à son cluster [11].

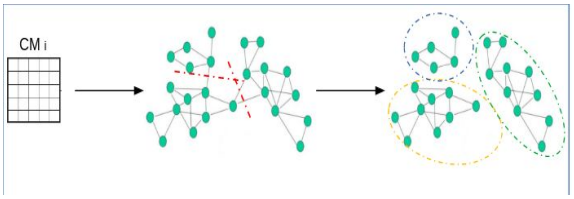

Figure. 5: l'algorithme de coupes normalisées est appliqué à chaque matrice de co-association générique

Cette méthode est proposée sous trois approches.

1. Le degré confiance moyenne d'affectation des objets aux clusters: cette première approche calcule pour chaque objet xi le degré de confiance de son appartenance aux clusters:

$$AC(P^*) = \frac{1}{n}\sum_{i=1}^{n} \mathrm{conf}(\mathbf{x_i})$$

Où P* est la partition à évaluer, n le nombre d'objets et conf(xi) est calculé comme suit:

$$\mathrm{conf}(\mathbf{x}_i) = \left(\frac{1}{|C_{P_i}|-1}\sum_{j:\mathbf{x}_j\in\{C_{P_i}\}\setminus\mathbf{x}_i} C_{ij}\right) - \left(\max_{1\le k\le K, k\neq P_i}\frac{1}{|C_k|}\sum_{j:\mathbf{x}_j\in C_k} C_{ij}\right)$$

où |Cpi| correspond au nombre de clusters, et la valeur de confiance est comprise entre -1 et 1, plus il est proche de 1 plus l'objet xi est considéré comme bien classé, respectivement plus elle est proche de -1 plus l'objet Xi et considéré comme mal classé, donc la meilleure partition correspond à celle dont le degré de

confiance moyenne de ces objets est la plus élevée.

2. Le degré de confiance moyenne de m plus proches voisins: la deuxième approche est basée sur les m plus proche voisin, ici au lieu de calculer la moyenne des fréquences de xi avec tous les objets, seuls les m les plus proches du xi en matière de distance seront calculés, et m sera donné par l'utilisateur,

$$ANC(P^*,m)=\frac{1}{n}\sum_{i=1}^{n}\left(\frac{\sum_{j:\mathbf{x}_j\in V(\mathbf{x}_i,C_{P_i},m)} C_{ij}}{|V(\mathbf{x}_i,C_{P_i},m)|}-\max_{1\le k\le K,k\neq P_i}\frac{\sum_{j:\mathbf{x}_j\in V(\mathbf{x}_i,C_k,m)} C_{ij}}{|V(\mathbf{x}_i,C_{P_k},m)|}\right).$$

3. Le degré de confiance moyenne de m plus proches voisins avec m calculé dynamiquement: cette troisième approche est une extension de la seconde, au lieu d'utiliser la même valeur m pour tous les clusters, elle sera calculée dynamiquement comme suit:

$$m_i=\left\lceil \alpha \sum_{j\in\{1,\cdots,n\}\setminus i} C_{ij}\right\rceil,$$

avec α est donné par l'utilisateur.

4.Expérimentation :

4.1 Prétraitement de texte :

Ici nous allons exposer les différentes méthodes de prétraitement de texte que nous avons utilisé pour préparer notre dataset pour l'étape de clustering :

1. Tokenisation : étant donnée une séquence de caractères, et une unité de document définit, la tokenisation consiste à découper cette séquence en morceaux appelés token tout en éliminant en même temps certains caractères comme les ponctuations.

2. Élimination des stops word : cette étape consiste à éliminer les mots trop fréquents et qui ne porte pas de sens au texte comme les mots de liaisons.

3. Lemmatisation et racinisation : pour la lemmatisation il s'agit de transformer un token (mot) dans sa forme canonique (exp : mettre les verbes à l'infini, transformer un nom en forme masculin singulier…etc.). La racinisation consiste à relever les suffixes et les préfixes d'un mot pour ne garder que la racine (stem) de celui-ci.

4. Transformation par Tf-Idf : cette étape consiste à produire la matrice termes-document qui sera utilisé comme entrée pour notre le modèle de clustering, et elle comporte plusieurs étapes :

Calcule $tf_{t,d}$ (term frequency) : il s'agit de déterminer le nombre de fois où chaque terme *t* apparaît dans un document *d.*

Calcule de idf (inverse document frequency) :il s'agit ici de déterminer le pouvoir discriminant d'un terme dans un document cela en calculant la l'inverse de la fréquence documentaire du terme : $idf_t = N/n$ tel que N est le nombre de documents total de la collection et n est le nombre de documents qui contiennent le terme *t* dans la collection.

Calcul de tf-idf : pour cela il suffit de faire le produit entre le tf et idf : $w_{t,d} = tf \cdot idf$ où *w* sera le poid qu'on affectera pour chaque terme dans document à l'intérieur de la matrice termes-document.

4.2 Dataset :

Pour pouvoir tester les performances de notre dataset, nous avons choisi d'utiliser le dataset 20 newsgroup, qui est un dataset très utilisé dans pour tester les algorithmes de traitement de texte.
Les caractéristiques du dataset sont résumés dans le tableau suivant

Dataset	Nombre de catégorie	Taille du vocabulaire	Taille du corpus
20nwg	4	10000	3959

Table 1. Description du dataset utilisé

4.3 Résultats :

Pour mesurer les performances de notre approche nous avons utilisé les mesures de validation suivantes :

Adjusted-Mutual-Information (AMI) : cette mesure sert à déterminer à quel point les informations réelles sont représentées dans le résultat du clustering.

Homogeneity (HOM) : cette métrique détermine la mesure dans laquelle les clusters ne contiennent que les points qui sont membre d'une seule classe.

Completeness (COM) : cette métrique représente la mesure dans laquelle tous les points qui sont membres d'une classe donnée appartiennent au même cluster.

V-mesure : représente la moyenne harmonique entre l'Homgeneity et la Completeness.

Adjusted Rand Index (ARI) : cette métrique calcul la similarité entre deux partitions en comptant les pairs qui sont au même ou à des différents clusters dans les clusters réels et prédits.

Nous avons comparé notre algorithme à deux autres algorithmes de clustering populaires à savoir le DBSCAN ainsi que l'Agglomérative Clustering et on a obtenu les résultats suivants**:**

algo	AMI	HOM	COM	V-m	ARI
dbscan	.157	.192	.188	.190	.03
aglo-cl	.480	.476	.484	.480	.481
AdaEAC-m	.594	.527	.681	.594	.509

Table 2. Résultats obtenus

4.Conclusion :

Dans ce projet, nous avons eu l'occasion de travailler sur le traitement des données textuelles en utilisant un algorithme personnalisé. Cet algorithme appartient à la famille des algorithmes de l'ensemble clustering qui ont pour but d'améliorer les résultats de clustering habituels en générant plusieurs modèles de clustering et en utilisant une fonction consensus pour retrouver le modèle de clustering qui se rapprochera le plus des modèles générés et qui produira une possible partition de données meilleures que les modèles initiaux.
Nous avons eu l'occasion de tester cet algorithme sur des données textuelles , et nous avons constaté que cette implémentation de l'algorithme a donné des résultats assez intéressant en la comparant à certains modèles existants.
Enfin nous pour la suite de nos études, nous souhaitons continuer sur l'exploration sur les différentes approches utilisées pour le traitement de texte.

Références

[1] Joao M. M. Duarte, Ana L. N. Fred , and F. Jorge F. Duarte, Adaptive Evidence Accumulation Clustering Using the Confidence of the Objects' Assignments.

[2] Md Rashadul Hasan Rakib, Norbert Zeh, Magdalena Jankowska, Evangelos Milios, Enhancement of Short Text Clustering by Iterative Classification

[3] Juan Zamora ,Héctor Allend-cid ,Marcelo Mendoza, Distributed Clustering of Text Collections

[4] Minh D. Nguyen and Won-Yong Shin, An Improved Density-Based Approach to Spatio-Textual Clustering on Social Media

[5] S.Yang, G.Huang , B.Cai,

Discovering Topic Representative Terms for Short Text Clustering

[6] Jaromír Novotný and Pavel Ircing, Unsupervised Document Classification and Topic Detection.

[7] Amir Hadifar, Lucas Sterckx, Thomas Demeester, Chris Develder, A Self-Training Approach for Short Text Clustering.

[8] Xiaoye Tan, Rui Yan, Chongyang Tao, and Mingrui Wu, Classification Over Clustering: Augmenting Text Representation with Clusters Helps!

[9] Rasim M. Alguliyev, Ramiz M. Aliguliyev, Nijat R. Isazade, Asad Abdi and Norisma Idris, COSUM:Text summarization based on clustering and optimization.

[10] Junaid Rashid, Syed Muhammad Adnan Shah, Aun Irtaza, Toqeer Mahmood, Muhammad Wasif Nisar, Muhammad Shafiq , and Akber Gardez, Topic Modeling Technique for Text Mining Over Biomedical Text Corpora Through Hybrid Inverse Documents Frequency and Fuzzy K-Means Clustering.

[11] João M. M. Duarte, F. Jorge F. Duarte, Ana L. N. Fred, Adaptive Evidence Accumulation Clustering Using the Confidence of the Objects' Assignments