M2 MALIA-MIASHS :
projet Network Analysis for Information Retrieval (consignes)

Julien Velcin, Université Lyon 2, Laboratoire ERIC

Février 2024

L'objectif principal de ce projet est de développer une solution d'analyse d'un corpus structuré qui comporte plusieurs fonctionnalités :

- chargement rapide des données et affichage de quelques statistiques,
- visualisation du corpus pour donner une idée de la structure des données,
- inclusion d'un petit moteur de recherche permettant de faire des requêtes par mots clefs,
- nouvelle structuration des données à l'aide de techniques de clustering,
- classification supervisée des données en prenant en compte la structure et l'information textuelle.

Pour cela, il s'agit de mettre en pratique tout ou partie de ce que vous avez vu en cours. C'est ce projet, réalisé en binôme qui sera évalué, en complément de l'examen écrit final. Vous trouverez des éléments pour vous guider dans les prochains paragraphes.

Fonction 1 : Acquisition des données

Le jeu de données est issue d'un cas d'étude qui peut différer d'une année sur l'autre et d'un profil d'étudiant à l'autre. Il est important de récupérer, nettoyer et sauvegarder les données dans un format facilement réutilisable. Les données comportent nécessairement à minima deux types de champs : une information textuelle (par ex. le titre d'un article) et une information structurelle (qui permet de relier les textes ou les auteurs des textes). Il faut aussi vérifier qu'il est possible de mettre en place une tâche de classification automatique supervisée. A ce stade, vous pouvez calculer quelques statistiques simples : nombre de documents et taille moyenne, nombre d'auteurs, distribution des documents par auteur, distribution temporelle s'il y a lieu, nombre et distribution des classes.

Fonction 2 : Prise en compte de la structure du corpus

Vous avez à votre disposition des informations qui vous permettent de structurer votre corpus. Construisez un graphe à partir de ces informations en choisissant l'identité des nœuds (par ex. document ou auteur) et ce qui constitue le lien entre eux (par ex. lien de citation ou de co-autorat). Vous pouvez calculer de nouvelles statistiques basées sur la structure du graphe : distribution des degrés (ou des autres mesures de centralité), nombre de composantes connexes, largeur, densité, etc. Ces informations peuvent vous être utiles pour préparer le graphe aux traitements ultérieurs, en particulier réduire sa taille afin que les analyses soient réalisées dans des temps raisonnables. Pour finir, utilisez une technique de visualisation des données pour vous faire une idée de la topologie de vos données. N'hésitez pas à projeter certaines caractéristiques des nœuds (par ex. la classe) sur cette visualisation.

Fonction 3 : Moteur de recherche

L'une des fonctionnalités qui est demandé est de pouvoir entrer un à plusieurs mots-clefs et retourner à l'utilisateur un ensemble d'articles relatifs à cette requête. Pour commencer, la solution implémentée peut être de calculer une simple similarité basée sur les mots, ce qui nécessite d'indexer le corpus à l'aide d'un outil comme ceux fournis par `scikit-learn`. Une meilleure solution consiste à calculer une similarité sémantique entre la requête et le texte des documents, en employant des techniques de représentation du sens des phrases, tel que USE ou S-BERT. Pour finir, une solution intéressante et plus exploratoire consiste à prendre en compte non seulement l'information textuelle mais aussi les relations structurelles entre les documents et les auteurs. Il s'agit par ex. d'utiliser des techniques de plongement de graphes (comme les GNNs) pour calculer des représentations qui prennent en compte le voisinage des nœuds.

Fonction 4 : Ajout de clustering

Vous pouvez tester un (ou plusieurs) algorithmes de clustering classiques, comme le clustering spectral ou Louvain. Il existe bien sûr d'autres solutions, tel que les *block models*. Ajoutez ces catégories calculées automatiquement dans vos données. Vous pouvez ainsi faire en sorte de les projeter sur la visualisation. Il peut être intéressant d'analyser la différence entre la classification qui vous est donnée et celle-ci, soit de manière purement qualitative, soit en employant des mesures comme l'ARI ou l'AMI.

Fonction 5 : Classification supervisée

Vous devez résoudre un problème de classification des nœuds du graphe dans certaines catégories prévues à l'avance. Il s'agit par exemple de prédire la catégorie de la publication en fonction de son domaine (par ex. *machine learning* ou traitement d'image). Pour cela, vous pouvez utiliser les caractéristiques textuelles uniquement (par ex. issues d'une vectorisation sémantique), les caractéristiques structurelles (par ex. un vecteur de caractéristiques du nœud ou un vecteur issue de `Node2Vec`), ou les deux à la fois. Des expérimentations intéressantes consistent à comparer ces différents modes de représentation et d'observer si les erreurs sont les mêmes ou non, voire de trouver selon quelles conditions l'une est préférable à l'autre.

Ce qu'il faut rendre :

1. rapport sous la forme d'un article de maximum 8 pages (format Springer)
2. code que vous avez utilisé, abondamment commenté, qui permet de reproduire vos expérimentations

L'utilisation de ChatGPT, ou autre technologie approchant, est toléré à condition que vous indiquiez clairement les endroits où vous l'avez utilisé et comment.