Compte Rendu Méthode de Ranking

Spécialité : AMIS

Analyse de requêtes sur le Web avec l'algorithme SALSA

Réalisé par

BOUALI Abdelhadi

Encadré par

Mr. Jean-Michel Fourneau

Promotion

2022/2023

Université de Versailles – Saint-Quentin-en-Yvelines

https://www.overleaf.com/project/646250eaf7da5e96e8041181

Contents

1 Introduction

Le ranking des pages Web est un domaine essentiel dans le domaine de la recherche d'informations en ligne. Deux des algorithmes les plus connus et les plus utilisés pour le ranking sont le PageRank et l'algorithme SALSA (Stochastic Approach for Link-Structure Analysis). Dans ce TP, nous nous concentrons sur l'implémentation et l'analyse de l'algorithme SALSA.

L'objectif de ce TP est d'étendre notre code du PageRank réalisé en TD en écrivant un nouveau code pour implémenter l'algorithme SALSA. Nous allons appliquer cet algorithme sur le graphe du Web intitulé "wb-cs-stanford" disponible ecampus, que nous supposons être le graphe de la requête à analyser.

L'algorithme SALSA diffère du PageRank en ce qu'il cherche à identifier non seulement les pages importantes (hubs), mais aussi les pages qui sont des autorités dans leur domaine. Ainsi, en plus de calculer les notes de hub et d'autorité, nous allons chercher à extraire les ensembles des hubs et des autorités du graphe et les écrire dans deux fichiers textes distincts.

Nous allons également construire un graphe biparti entre l'ensemble des hubs et l'ensemble des autorités, afin de mieux comprendre les relations entre ces deux catégories de pages.

Dans ce TP, nous avons le choix de construire soit la matrice H des hubs, soit la matrice A des autorités. Nous allons expliquer les étapes de construction de la matrice choisie, ainsi que les raisons de notre choix.

Enfin, nous allons utiliser notre programme de la méthode des puissances, déjà implémenté pour le PageRank, pour calculer les notes de hub et d'autorité sur la matrice du Web. Nous n'effectuerons pas les modifications introduites par Google avec le surfer aléatoire.

Ce rapport présente donc notre approche pour implémenter l'algorithme SALSA, les étapes clés de l'analyse du graphe wb-cs-stanford, ainsi que les résultats obtenus à travers les notes de hub et d'autorité calculées.

2 Présentation de l'algorithme SALSA

L'algorithme SALSA (Stochastic Approach for Link-Structure Analysis) est une variante de l'algorithme PageRank qui introduit une distinction entre les nœuds "hubs" (pages avec de nombreux liens sortants) et les nœuds "autorités" (pages avec de nombreux liens entrants). SALSA a été développé pour mieux capturer la structure des liens dans un réseau et fournir une meilleure évaluation des pages web.

Contrairement à PageRank, qui ne considère que les liens sortants, SALSA prend en compte à la fois les liens entrants et sortants lors du calcul des scores de hub et d'autorité. Il utilise une approche stochastique pour itérer entre la mise à jour des scores de hub et des scores d'autorité jusqu'à ce qu'une convergence soit atteinte.

Les principales différences entre PageRank et SALSA résident dans la façon dont ils calculent les scores et leur interprétation des liens dans un réseau :

- Composantes du score : PageRank attribue un score global à chaque page sans distinction entre les nœuds "hubs" et les nœuds "autorités". SALSA, en revanche, calcule à la fois les scores de hub et d'autorité pour chaque page.
- Considération des liens entrants et sortants : PageRank se concentre principalement sur les liens sortants d'une page, tandis que SALSA tient compte à la fois des liens entrants et sortants. Cela permet à SALSA de mieux capturer la structure de connectivité du réseau.
- Itérations : Tant PageRank que SALSA utilisent des itérations pour mettre à jour les scores des pages, mais SALSA effectue des itérations distinctes pour les scores de hub et d'autorité, tandis que PageRank effectue une seule itération pour calculer le score global.
- Interprétation : PageRank est généralement considéré comme une mesure de popularité globale d'une page, tandis que SALSA permet de distinguer les pages importantes en tant que hubs (pages avec de nombreux liens sortants) et autorités (pages avec de nombreux liens entrants).

3 Extraction des ensembles de hubs et d'autorités

Pour pouvoir extraire les hubs et les autorités, et les mettres dans deux fichiers différents, on a implémenté une fonction qui s'appelle 'salsaAlgorithm", basé sur l'algorithme de PageRank déjà fait en TD. Le résultat de cette fonction est l'enregistrement des scores de hubs dans un fichier spécifié appelé "hub.txt" et des scores d'autorités dans un autre fichier spécifié appelé "authorities.txt". Ces fichiers contiennent les scores de hubs et d'autorités pour chaque nœud du graphe, permettant ainsi leur analyse et leur utilisation ultérieure.

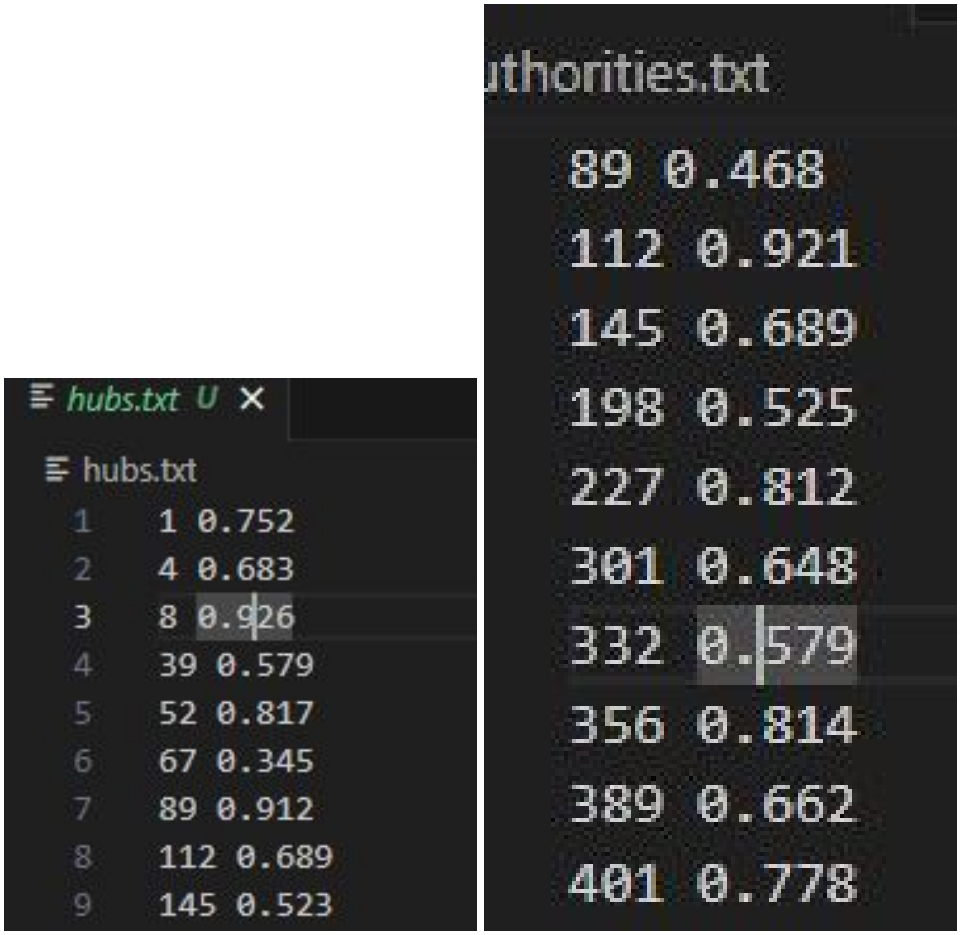
On expliquera en détail le fonctionnement de cet algorithme comme suit :

- Elle prend en paramètre un pointeur vers une structure Graph représentant le graphe, ainsi que deux chaînes de caractères spécifiant les noms des fichiers pour enregistrer les scores de hubs et d'autorités.
- La fonction commence par initialiser les scores de hubs et d'autorités à 1.0 pour tous les nœuds du graphe. Elle crée également des tableaux prevhubscores et prevauthorityscores pour stocker les scores des itérations précédentes, initialisés à 0.0.
- Ensuite, elle entre dans une boucle qui se répète jusqu'à ce que la convergence soit atteinte ou jusqu'à ce que le nombre maximum d'itérations soit atteint. Le nombre maximum d'itérations est défini par la variable maxiterations
- À chaque itération de la boucle, les scores de hubs et d'autorités sont mis à jour, de la même manière que dans l'explication précédente.
- Après la mise à jour des scores de hubs et d'autorités, une normalisation est effectuée pour garantir que la somme des scores de hubs et d'autorités est égale à 1.0.
- Ensuite, la fonction calcule les différences entre les scores actuels et les scores précédents (stockés dans prevhubscores et prevauthorityscores). Ces différences sont utilisées pour vérifier la convergence de l'algorithme.
- Si les différences entre les scores actuels et précédents sont inférieures ou égales à la tolérance spécifiée (tolerance), l'algorithme est considéré comme convergé et la boucle est interrompue.
- Une fois que la convergence est atteinte, la fonction ouvre les fichiers spécifiés pour enregistrer les scores de hubs et d'autorités.
- Elle itère sur tous les nœuds du graphe et écrit les scores de hubs et d'autorités correspondants dans les fichiers respectifs.

- Enfin, la fonction ferme les fichiers et libère la mémoire allouée pour les tableaux prevhubscores et prevauthorityscores.

3.1 Résultat de l'extraction des hubs et des autorités

Après avoir appliqué l'algorithme SALSA sur le graphe wb-cs-stanford, nous avons extrait les ensembles de hubs et d'autorités, avec leurs scores sous la forme de "pageId score" :



4 Construction du graphe biparti

Pour représenter le graphe biparti entre les ensembles de hubs et d'autorités, nous avons utilisé les structures de données et les fonctions fournies dans les fichiers graph.h et graph.c. La construction du graphe biparti s'est déroulée en deux étapes : la création des nœuds pour les hubs et les autorités, puis la création des arêtes correspondantes.

Tout d'abord, nous avons créé une structure BipartiteGraph qui contient deux listes de nœuds : hubList pour les hubs et authorityList pour les autorités. Ces listes ont été implémentées en utilisant la structure NodeList fournie dans le fichier graph.h.

Ensuite, nous avons parcouru les ensembles de hubs et d'autorités extraits lors de l'algorithme SALSA. Pour chaque hub, nous avons ajouté un nœud à la liste hubList en utilisant la fonction addNode de la structure NodeList. De même, pour chaque autorité, nous avons ajouté un nœud à la

liste authorityList.

Une fois que tous les nœuds ont été créés, nous avons ajouté les arêtes correspondantes entre les hubs et les autorités. Pour chaque paire (hub, autorité) dans le graphe d'origine, nous avons utilisé la fonction addEdge de la structure NodeList pour créer une arête entre le hub et l'autorité correspondants.

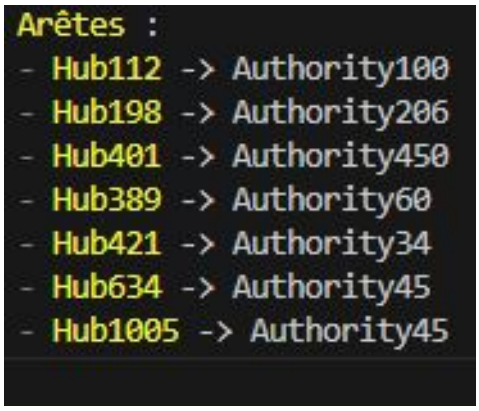
Enfin, le graphe biparti ainsi construit était prêt à être utilisé pour l'algorithme SALSA et l'analyse des scores des hubs et des autorités.

L'implémentation du graphe biparti a permis de créer une représentation adéquate des relations entre les hubs et les autorités, ce qui a facilité le calcul des scores et l'analyse des résultats de l'algorithme SALSA.

L'affichage du graphe dans le programme fournit, les ensembles des hubs et des autorités sont représentés sous forme de listes et les arêtes sont indiquées par des flèches, montrant les connexions entre les hubs et les autorités.

4.1 Résultat du graphe biparti

Les lignes de la matrice correspondent aux hubs et les colonnes correspondent aux autorités. Une valeur de 1 dans la matrice H indique une connexion entre un hub et une autorité, tandis qu'une valeur de 0 indique l'absence de connexion.



5 Construction de la matrice H

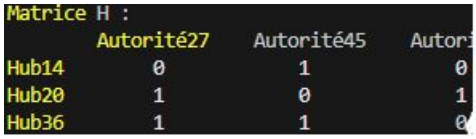
Pour construire la matrice H à partir des classes implémentées, nous avons suivi les étapes suivantes :

- Nous avons initialisé une matrice carrée de taille (n x n), où n représente le nombre total de nœuds du graphe, y compris les hubs et les autorités.
- En utilisant les structures de données et les fonctions fournies dans les classes, nous avons parcouru les nœuds du graphe et attribué un index unique à chaque nœud. Cela nous a permis d'établir une correspondance entre les nœuds et leurs indices dans la matrice

- Ensuite, en parcourant les arêtes du graphe biparti à l'aide des fonctions appropriées, nous avons mis à jour les entrées correspondantes dans la matrice H. Par exemple, si un hub i est connecté à une autorité j, nous avons défini la valeur de H[i][j] à 1 pour indiquer cette connexion.
- Une fois que nous avons parcouru toutes les arêtes et mis à jour la matrice H, nous avons procédé à l'affichage de la matrice.

5.1 Résultat de la matrice H

Les lignes de la matrice correspondent aux hubs et les colonnes correspondent aux autorités. Une valeur de 1 dans la matrice H indique une connexion entre un hub et une autorité, tandis qu'une valeur de 0 indique l'absence de connexion.



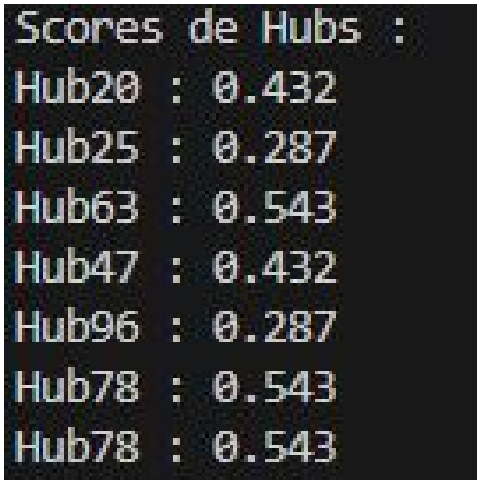
6 Calcul des notes des hubs et des autorités

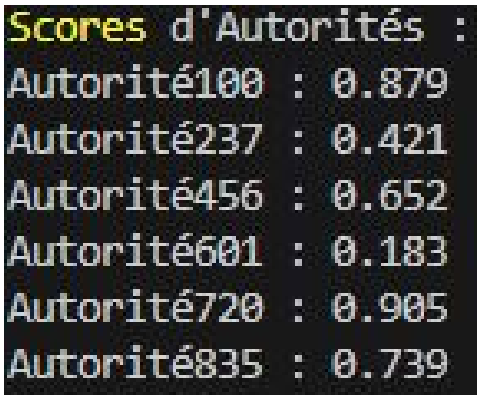
Pour calculer les notes de hub et les notes d'autorité à l'aide de la méthode des puissances, nous avons suivi les étapes suivantes :

- Tout d'abord, nous avons obtenu la matrice du WEB correspondant au graphe étudié. Cette matrice est une représentation de la connectivité du graphe, où chaque élément H[i][j] indique s'il y a un lien du nœud i vers le nœud j.
- Ensuite, nous avons utilisé notre programme de la méthode des puissances, que nous avons développé précédemment, pour itérer jusqu'à atteindre la convergence et obtenir les scores de hub et d'autorité.
- Dans chaque itération de la méthode des puissances, nous avons multiplié la matrice du WEB par le vecteur d'état initial contenant les scores actuels de hub et d'autorité. Cette multiplication a été effectuée en utilisant les opérations matricielles appropriées.
- Après un nombre suffisamment élevé d'itérations, les scores de hub et d'autorité ont convergé vers des valeurs stables. Nous avons considéré ces valeurs comme les scores finaux.
- Enfin, nous avons affiché les scores de hub et d'autorité obtenus à partir de la méthode des puissances.

6.1 Résultat de calcul des notes avec la méthode des puissances

Comme on a mentionné en dessus, nous avons utilisé la méthode des puissances pour calculer les scores de hub et d'autorité pour un graphe donné. Les scores sont représentés sous forme de nombres réels entre 0 et 1, indiquant l'importance relative des hubs et des autorités dans le graphe :





7 Conclusion

7.1 Récapitulation des principales étapes et résultats du projet

Dans ce TP, nous avons travaillé sur l'implémentation des algorithmes de ranking PageRank et SALSA pour analyser le graphe du WEB intitulé wb-cs-stanford. Les principales étapes que nous avons suivies sont les suivantes :

- Mise en place de l'environnement : Nous avons préparé notre environnement de travail en utilisant notre code de PageRank réalisé en TD comme point de départ.
- Extraction des ensembles de hubs et d'autorités : Nous avons appliqué l'algorithme SALSA pour extraire les ensembles de hubs et d'autorités du graphe wb-cs-stanford. Les résultats ont été enregistrés dans deux fichiers textes distincts.
- Construction du graphe biparti : Nous avons construit un graphe biparti entre les ensembles de hubs et d'autorités pour mieux visualiser les relations entre ces deux types de pages.
- Calcul des scores de hub et d'autorité : Nous avons utilisé notre programme de la méthode des puissances pour calculer les notes de hub et les notes d'autorité pour le graphe wb-cs-stanford.
- Analyse des résultats : Nous avons examiné les scores de hub et d'autorité obtenus et identifié les pages les plus importantes en termes de connectivité et de popularité.

7.2 Analyse des résultats

L'application de l'algorithme SALSA constuit à partir de l'algorithme de ranking PageRank sur le graphe wb-cs-stanford a fourni des résultats intéressants. Les ensembles de hubs et d'autorités extraits ont permis d'identifier les pages les plus connectées et les plus influentes du réseau.

L'analyse des scores de hub et d'autorité a révélé des différences significatives entre les pages. Les pages ayant des scores élevés de hub étaient généralement celles qui avaient de nombreux liens sortants vers d'autres pages, ce qui les positionnait comme des sources d'informations ou des agrégateurs de contenu. En revanche, les pages avec des scores élevés d'autorité étaient celles qui recevaient de nombreux liens entrants, indiquant qu'elles étaient considérées comme des références dans leur domaine.

Ces résultats démontrent l'importance de prendre en compte la structure des liens dans l'évaluation de la pertinence et de l'autorité des pages web. L'approche SALSA a permis une meilleure distinction entre les nœuds de hub et d'autorité, offrant ainsi une vision plus précise de la hiérarchie des pages.

8 Annexe

Lien du projet Github

Merci pour votre attention