

Projet SINF1250

« Ranking de réseaux sociaux et pages web: PageRank »

| | |
|-------------------|---|
| Professeur | Marco Saerens marco.saerens@uclouvain.be |
| Téléphone | 010 47 92 46 |
| Bureau | b.108 |
| Adresse | Université catholique de Louvain Place des Doyens 1 1348 Louvain-La-Neuve Belgique |
| Assistant | Mathieu Zen et Bertrand Lebichot mathieu.zen@uclouvain.be |
| Bureau | b.102 |
| Date | 15 Mars 2016 |

Objectif : Le but est d'implémenter et de tester un algorithme permettant de classer (rank) les noeuds d'un graphe en assignant un score d'importance à chaque noeud. Cet algorithme est celui de PageRank avec téléportation et vecteur de personnalisation. Vous travaillerez par groupe de deux étudiants (merci de vous inscrire dans un groupe sur iCampus) et l'implémentation se fera en java.

Il faut suivre rigoureusement cet algorithme (pas de fantaisies svp).

Vous trouverez toutes les informations utiles dans les diapos du cours mais aussi dans des ouvrages de "link analysis" (par exemple l'ouvrage *Google's PageRank and Beyond* de Amy Langville et Carl Meyer).

Méthode Java :

- **Input :** Une matrice d'adjacence A d'un graphe dirigé, pondéré et régulier G ainsi qu'un paramètre de téléportation α entre 0 et 1 (inutile de le vérifier) et enfin un vecteur de personnalisation q .
- **Output :** Un vecteur x contenant les scores d'importance des noeuds ordonnés dans le même ordre que la matrice d'adjacence.

En plus, nous vous demandons d'écrire un script "main" en Java (dans le même fichier ou directory que la procédure qui calcule le score PageRank) qui charge le fichier de données (la matrice d'adjacence) et qui appelle la procédure calculant les scores PageRank, et les imprime à l'écran. Cette fonction sera exécutée pour vérifier que votre programme fonctionne bien.

Vous pouvez calculer ces scores PageRank en utilisant la techniques que vous voulez : résoudre un système d'équations linéaires par une méthode itérative, utiliser la power method pour calculer le vecteur propre de gauche de la matrice de transition associé à $\lambda = 1$, etc.

Notez que vous pouvez utiliser une librairie Java externe de calcul et de manipulation matricielle/vectorielle. Cette librairie vous évitera d'implémenter la multiplication matricielle de manière à vous concentrer sur l'algorithme PageRank. Par contre, évitez d'utiliser les fonctions permettant de calculer des vecteurs propres ou résoudre des systèmes d'équation linéaire car le but du projet est justement de comprendre et d'implémenter cela ! Donc, vous ne pouvez utiliser cette librairie que pour effectuer des opérations matricielles simples telles que la transposée, la multiplication matricielle, etc.

Données : Deux fichiers texte contenant les matrices d'adjacence de graphes de petits réseaux sociaux sont disponibles sur iCampus. Ces matrices contiennent les affinités a_{ij} entre les noeuds i et j si les deux noeuds sont directement connectés et 0 si il n'y a pas d'arc reliant i à j . Ces données pourront servir pour tester votre programme et comparer les résultats avec les autres groupes.

Attention, actuellement les pondérations a_{ij} de la matrice d'adjacence représentent des affinités entres noeuds. Il faudra donc les transformer en probabilités de transition.

Rapport : Le rapport est un fichier PDF (6 pages maximum ; de préférences utiliser Latex) qui se compose de :

- Une explication de (1) votre procédure Java, (2) l'algorithme utilisé pour calculer les scores PageRank, (3) la librairie de calcul matriciel utilisée, ainsi que (4) la méthode mathématique/numérique utilisée pour déterminer les scores (avec les formules mathématiques).
- En annexe les scores PageRank sans téléportation et $\alpha = 1$ pour les deux graphes test.
- Le code complet en annexe – n'oubliez pas de bien commenter le code !

Langage de programmation : L'implémentation devra impérativement être codée en Java mais vous pouvez vous entraîner en Matlab, Octave ou Python avant l'implémentation Java pour bien comprendre le programme (prototypage précédant l'implémentation).

Evaluation et consignes : Le projet est à réaliser par groupes de deux étudiants. L'évaluation portera sur le contenu du rapport (maximum 6 pages) et le code (lisibilité, structure, commentaires,...) et comptera pour deux points sur 20. Les fichiers, c'est-à-dire l'unique fichier de code source et le rapport pdf, sont à remettre sur iCampus au plus tard le vendredi 13 mai (dernier jour de cours) avant 24h00. La note sera la même pour tous les membres du groupe.

Bon courage !
