Dr. Laura Elena RAILEANU

Florian POULIN

Analyse des hyperliens

Laboratoire Nº 6

A. Objectifs

Dans ce laboratoire, vous allez implémenter l'analyse des hyperliens et l'intégrer dans votre moteur de recherche. Plus précisément, les points étudiés dans ce laboratoire seront :

- Le Couplage Bibliographique et la Co-Citation
- Le Hub et l'Autorité
- Le PageRank

B. Références

Cours «Recherche d'Information Multimédia » de Laura Raileanu.

C. Rapport

A remettre : Un rapport comportant les objectifs, la description des démarches adoptées et une conclusion personnelle. Vos codes sources, en Java.

A remettre : Au plus tard à la fin de la dernière séance du trimestre.

D. Donnée

Ce laboratoire se déroule en deux étapes. Dans la première, vous implémenterez l'analyse des liens et dans la seconde vous intégrerez cette analyse à votre moteur de recherche.

1. Implémentation de l'analyse des hyperliens

Soit :

- G = (V, E) un graphe représentant un ensemble de sites, avec V = (p_1, p_2, …, p_n)
- M la matrice d'adjacence de G, avec $m_{i,k} = 1$ si $(p_i,p_k) \in E$, 0 autrement

On peut calculer le Couplage bibliographique (BC), la Co-Citation (CC), le Hub (h), l'Autorité (a), et le PageRank (pr) comme:

$$BC = M \cdot M^T$$
$$CC = M^T \cdot M$$
$$h^c = M \cdot a^{c-1}$$
$$a^c = M^T \cdot h^{c-1}$$
$$\mathbf{Pr^c = normalize\ (0.85 * (M^T * pr^{c-1}) + E)}$$

Avec (pour le PageRank) :

M → Matrice d'adjacence

E → Vecteur de probabilités de saut aléatoire : others ➔ 0.15 / |V|

Pr^0 → others ➔ 1 / |V|

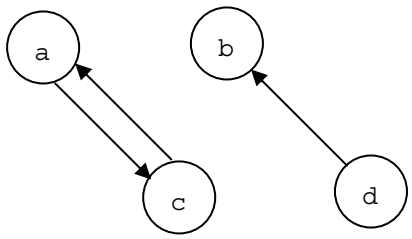
Dans cette première partie il vous est demandé d'implémenter ces 5 algorithmes. Pour vous aider, plusieurs fichiers Java sont mis à votre disposition à l'emplacement habituel. Le travail consiste à compléter la classe `LinkAnalysis` (5 méthodes à implémenter). Pour ce faire, vous avez à disposition les méthodes définies dans l'interface `AdjacencyMatrix`.

Pour tester votre travail, vous devez vous aider de la classe `GraphFileReader` qui vous permet d'obtenir une instance de matrice d'adjacence (un objet de type `AdjacencyMatrix`) d'un graphe défini dans un fichier.

Le fichier doit impérativement respecter le format de l'exemple suivant :

```
# nodes
a
b
c
d
# edges
a;c
c;a
d;b
```

(Fichier de graphe)

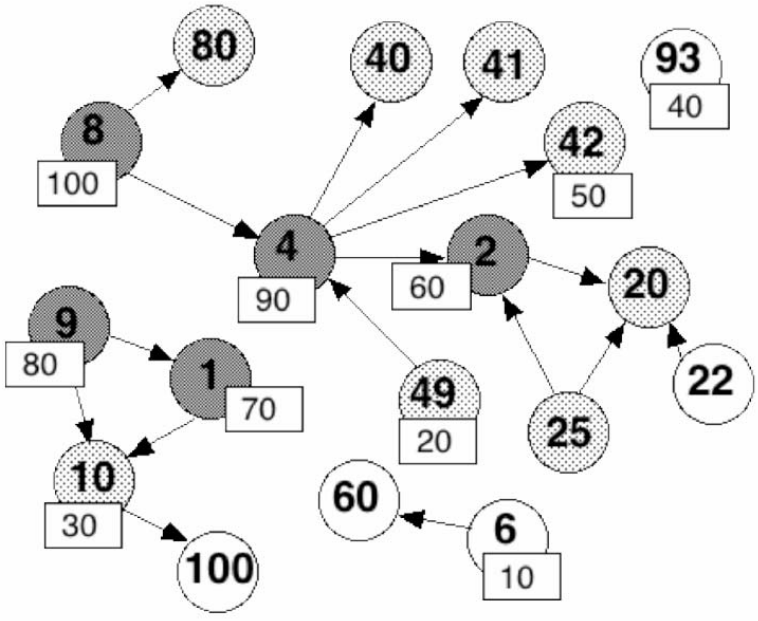


(Graphe résultant)

Votre programme devra **effectuer une analyse du graphe fourni ci-dessous**, et **afficher sur la console ou dans un fichier** (pas uniquement dans le rapport) **les informations suivantes** :

- Le couplage bibliographique entre les documents 25 et 22
- La valeur de co-citation entre les documents 20 et 2
- Les informations suivantes, pour chaque nœud :

  (Effectuez 5 itérations en normalisant à chaque étape)
  - Le Hub
  - L'autorité
  - Le PageRank



2. Intégration dans le moteur de recherche

La seconde partie du travail consiste à **intégrer l'analyse des liens dans votre moteur de recherche**. Pour ce faire, il vous faut générer une matrice d'adjacence représentant votre corpus en utilisant la classe `ArrayListMatrix`. Pour vous faciliter la tâche, inspirez-vous du code de `GraphFileReader`.

Ensuite, vous pourrez réutiliser le code développé en première partie pour connaître notamment le PageRank de chaque page. Finalement, **utilisez le PageRank des pages pour ordonner les résultats retournés par votre moteur de recherche** (par ordre décroissant de PageRank).

E. Indications

- Des classes vont sont fournies pour ce laboratoire afin de vous faciliter la tâche et vous donner un exemple d'implémentation. Si vous constatez des manques ou des erreurs dans ces classes, vous êtes autorisés à les modifier (voire même à les remplacer). Faites-en part à l'assistant ou au professeur en cas de modification, et parlez-en dans votre rapport ;
- Pour comprendre comment se servir correctement des classes fournies, référez-vous à leur javadoc (dossier `doc`) ;
- En cas de doute, n'hésitez pas à poser des questions à l'assistant ou au professeur.