Université de Versailles Saint-Quentin-en-Yvelines

# Compte Rendu Méthode de Ranking

# Calcul du PageRank par l'algorithme utilisant le Vecteur $\nabla$

Membres du groupe :

- AOUCHICHE Salah
- ALLAOUA MELISSA
- BELHANNACHI Moundji-Hocine
- CHAABANI Ahmed
- HADJ MAHFOUD Kenza
- HARAL Dylan

Mai 2022

# Table des matières

# 1 INTRODUCTION

Dans le cadre de notre projet nous sommes amenés à implémenter l'algorithme PageRank en utilisant le vecteur ∇ et de faire une comparaison avec l'algorithme de base sur des graphes du web par rapport aux performances, efficacité et temps d'exécution.

L'algorithme initial est le PageRank, lors du calcul de graphiques Web, ou plus précisément de matrices creuses, l'algorithme utilise un vecteur stochastique de la taille de la matrice. Il est initialisé et la matrice est multipliée en quelques itérations jusqu'à ce que ce vecteur stochastique converge. Le second algorithme est une modification de l'algorithme de PageRank. En effet, cet algorithme utilise deux vecteurs nabla et delta où l'initialisation de chaque case de nabla est le minimum de chaque colonne de la matrice G et l'initialisation de chaque case de delta est le maximum de chaque colonne de la matrice G.

# 2 Définition de l'algorithme

$$\nabla[j] = \min_i G\,[i, j]$$

$$\Delta[j] = \max_i G\,[i, j]$$

I. Initialiser $X^0 = \nabla$. $Y^0 = \Delta$

II. Faire une boucle sur k
(a) $X^{k+1} = \max\,(X^k, X^kG + \nabla\,(1 - \|X^k\|1)$

(b) $Y^{k+1} = \min\,(Y^k, Y^kG + \nabla\,(1 - \|Y^k\|1)$

III. Jusqu'à ce que $\|Xk - Y\,k\|1 < e$.

# 3 Graphe du web

On prend en lecture des fichiers textes avec les informations suivantes :

- Première ligne on trouve le nombre de sommets du graphe.
- Deuxième ligne on trouve le nombre d'arcs du graphe.
- A partir de là sur chaque ligne on trouve le numéro de sommet, le degré sortant de ce sommet et des couples (destination, cout).

# 4 Calculs et optimisations

Le calcul de ∇ et Δ se fait au même temps que la lecture du fichier dont la matrice est stockée.

## 4.1 Calcul de ∇

$$\nabla[j] = \min_i G[i, j]$$

On a $G[i, j] = \alpha P[i,j] + \frac{alpha}{N} f^t e + \frac{1-alpha}{N} e^t e$

Donc : $\min_i G[i, j] = \min_i (\alpha P[i,j] + \frac{alpha}{N} f^t e + \frac{1-alpha}{N} e^t e)$

- $\min_i(\alpha P[i, j]) = 0$ car la matrice ne contiens pas de boucle donc la diagonal est toujours à 0.

- $\min_i(\frac{alpha}{N} f^t e) = 0$ car on a au moins un sommet de degré sortant non nul (il n'y a pas de matrice nulle ) donc on aura au moins une valeur à 0 dans le vecteur f ce qui implique qu'on aura au moins une ligne à 0 dans la matrice $(\frac{alpha}{N} f^t e)$ .

- $\min_i(\frac{1-alpha}{N} e^t e) = [\frac{1-alpha}{N}, \frac{1-alpha}{N}, \ldots\ldots\ldots\ldots, \frac{1-alpha}{N}]$ car le résultat de $(\frac{1-alpha}{N} e^t e)$ est une matrice dont toutes les cases sont égales à $\frac{1-alpha}{N}$ .

Donc le vecteur ∇ sera toujours initialiser à :

$$\nabla [j] = \left( \frac{1-alpha}{N}, \frac{1-alpha}{N}, ……………, \frac{1-alpha}{N} \right)$$

## 4.2 Calcul de Δ

$$\Delta [j] = max_i\ G [i, j]$$

On a $$G[i, j] = \alpha P[i,j] + \frac{alpha}{N} f^{t} e + \frac{1-alpha}{N} e^{t} e$$

Donc : $$max_i\ G[i, j] = max_i (\alpha P[i,j] + \frac{alpha}{N} f^{t} e + \frac{1-alpha}{N} e^{t} e)$$

**Remarque** : On peut avoir des sommets de degré nul donc $max_i (\alpha P[i,j]) < \frac{1}{N}$ . On doit éviter ce cas donc à chaque fois qu'on trouve un sommet de degré nul on met le maximum de sa colonne à $\frac{1}{N}$ .

### 4.3 Calcul de l'algorithme

Maintenant qu'on a les vecteurs initiaux nabla et delta, on peut appliquer l'algorithme.

#### 4.3.1 Calcul de $X^{k+1} = max (X^k, X^kG + \nabla (1 - ||Xk ||1)$

$$X^0G = X^0 * \nabla \alpha P + (\frac{\alpha}{N} \nabla f^t + \frac{1-\alpha}{N} \nabla e^t) e$$

Terme1 = $X^0 * \nabla\ \alpha P$ (Vecteur)

Terme2 = $(\frac{\alpha}{N} \nabla f^t + \frac{1-\alpha}{N} \nabla e^t) e$ (Vecteur)

Terme 3 = $\nabla (1 - ||X^k ||1)$ (Vecteur)

$X^{k+1} = max (X^k$, Terme1 + Terme2 + Terme3)

#### 4.3.2 Calcul de $Yk+1 = min (Y^k, YkG + \nabla (1 - ||Yk ||1)$

$$Y^0G = Y^0 * \nabla \alpha P + \left(\frac{\alpha}{N} \nabla f^t + \frac{1-\alpha}{N} \nabla e^t\right) e$$

Terme1 = $Y^0 * \nabla \alpha P$ (Vecteur)

Terme2 = $\left(\frac{\alpha}{N} \nabla f^t + \frac{1-\alpha}{N} \nabla e^t\right) e$ (Vecteur)

Terme 3 = $\nabla (1 - ||Y^k||1)$ (Vecteur)

Yk+1 = min ($Y^k$, Terme1 + Terme2 + Terme3)

# 5 Résultats

## 5.1 Comparaison par rapport aux nombres d'itérations

| Matrice du Web | Taille de la matrice | Nombre d'arcs |
|---|---|---|
| wb-cs-stanford | 9914 | 36854 |
| Stanford | 281 903 | 2 312 497 |
| StanfordBerkeley | 68344 | 7 583 376 |
| Inde | 1 382 908 | 1 691 7053 |
| Web edu | 9 845 725 | 57 156 537 |
| Wikipedia2005 | 1 634 989 | 19 753 078 |
| Test | 6 | 16 |

Table 1: Données des Graphes du Web

| Matrice du Web | Nombre itération nabla | Nb itération PageRank |
|---|---|---|
| wb-cs-stanford | 130 | 55 |
| Stanford | 152 | 65 |
| StanfordBerkeley | 155 | 64 |
| Inde | 154 | 62 |
| Web edu | 167 | 61 |
| Wikepedia2005 | 146 | 35 |
| Test | 89 | 12 |

On constate que en exécutant l'algorithme de PageRank on trouve toujours des résultats nettement inférieurs à ceux de l'algorithme PageRank utilisant ∇. Et cela est dû au faits que l'algorithme de PageRank utilisant nabla doit faire en plus des calculs du produit vecteur*matrice et produit scalaire à chaque itération, l'algorithme va à chaque fois calculer le $X^k$ et $Y^k$ ce qui augmente la complexité et utilise d'avantage l'espace mémoire.

### 5.2 Comparaison par rapport aux Temps d'exécution (s)

| Matrice du Web | Temps nabla | Temps PageRank |
|---|---|---|
| wb-cs-stanford | 0.125973 (s) | 0.023727 (s) |
| Stanford | 53.466839 (s) | 10.783187 (s) |
| Inde | 52.098653 (s) | 19.727274 (s) |
| Web edu | 167 (s) | 61 (s) |
| Wikepedia2005 | 105.444817 (s) | 861.486755 (s) |
| Test | $40 \times 10^{-6}$ (s) | $7 \times 10^{-6}$ (s) |

On remarque que c'est les mêmes résultats que la comparaison par rapport aux nombres d'itérations.

## 6 Conclusion

On conclue que l'algorithme de PageRank est plus efficace que l'algorithme de PageRank en utilisant les deux vecteurs nabla et delta car la différence entre les deux algorithmes est importante, à chaque fois que le nombres de sommets ou le nombre d'arcs augmentent.

## 7 Annexe :

Lien GitHub : https://github.com/uvsq21921706/Projet_Ranking