AMD – Prof. Matchiodi (Massive Datasets)

UNIVERSITÀ DEGLI STUDI DI MILANO

M. Besim SAKAOGLU 985861

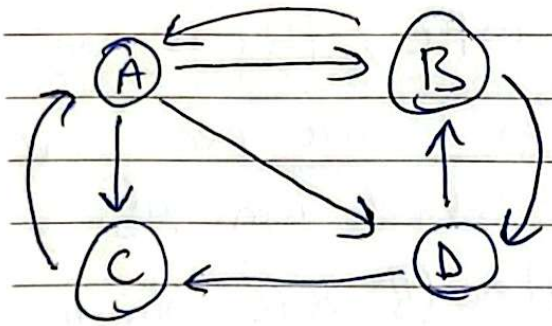
LINK ANALYSIS

I do have a relation which is a subset of 'Cartesian Product' btw 2 different sets.

PAGERANK algorithm: finding out the importance of webpage

one of the (2 founders of 'Google' & Larry Page)

↳ They proposed specific algorithm that tried to find out among the huge number of pages which are typically returned when you query something on the web ... We are not going to 2nd google page anymore ↳ showing the first most reliable results of Google



→ This graph as description of a portion of the web (very small one)

→ each node of the graph refers to a webpage

→ Edges are Directed & refers hyperlinks between pages (bidirectional)

→ Directed Edges because links in webpages are not bi-directional

→ The number of webpages estimated as $n \approx 10^9$ order of magnitude ten raised to nine

→ There's no universal definition of 'Big Data'. We deal with Big Data when we have an amount of data to be processed which is radically different from what we called small data approach. So, Big Data for Medicine field is radically different from Big Data in fields of Humanities or Computer Science etc. Especially, for what concerns Computer Science, we speak about Big Data, when we cannot reasonably use a single computer! (process or even store the data)

↳ So, if I do have n total webpages & I'll deal with a matrix $M = [m_{ij}]_{n \times n}$ square matrix 'n' times 'n'

→ The Generic entry 'M_{ij}' will tell me something about the relation that exist btw 2 of these webpages ('i' and j')
If 'n' is the order of '10^9', M will have 10^{18} elements (since it's a square Matrix). Let's do the Math, and check how much space I need in order to fully store this Matrix.

→ Before 'Google', on the search engines it was easy to cheat for showing their webpages at top since those relied on content. After 'Google', they changed 'notion of importance' from content to 'structure'. Relies on the structure of the web, hyperlinks from one page to another one. Webpage is important if it is linked by important webpages! OFC, this is not acceptable definition! It leads to an infinite recursion.

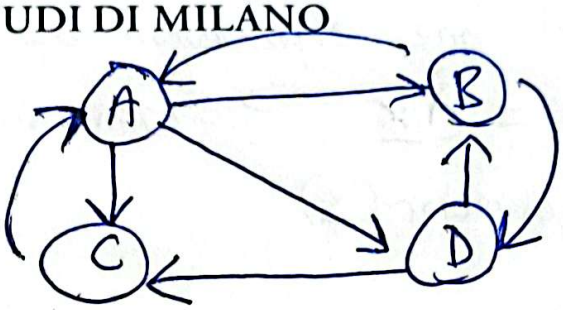
→ webpage 'A' (A) = noktasının önemine bakmak için A'ya gelen linklere bakıp (B) ve (C)'yi bulduk. Ama (B) ve (C)'nin important olup olmadığına bakmak için aynı process sonsuz kez tekrarla (infinite recursion olur!)

→ Basic Idea, 'Random Surfing': Just figured out that I have a population of huge number of individuals and I placed uniformly individuals in the various webpages and each individual will uniformly select at random one of the outgoing links that start from the page where he/she sits! So, we have sort of 'Discrete-time' process, at each time, all individuals select uniformly at random one of the outgoing links and follow it. And we will iterate this process. (Stochastic Process). Also, we want to figure out if there is some 'convergence' in this process. Ideally, if I have a convergence of the probability distribution of individuals to webpages

that prob. dist will identify the importance I'm seeking!

UNIVERSITÀ DEGLI STUDI DI MILANO

Mathematically:



CONNECTION MATRIX

$M = [m_{ij}]_{n \times n} \Longrightarrow$ A B C D — Source & row-wise

'STOCHASTIC BY COLUMN'

$$M = \begin{pmatrix} 0 & 1/2 & 1 & 0 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 1/2 & 0 & 0 \end{pmatrix} \begin{matrix} A \\ B \\ C \\ D \end{matrix}$$

Ex: $\text{Col}(A) = 0 + \frac{1}{3} + \frac{1}{3} + \frac{1}{3} = 1$ column 'A'

$m_{AA} = 0$ Gives self-loop yok

$m_{BA} = 1/2$ Gives B'nin out-degreesi 2

Destination & column-wise

$M_{ij} = \mathbb{P}(j \to i)$

$V_j(t) = \mathbb{P}(\text{sitting in } j \text{ at time } t)$

$V_j(t+1) = \sum_i \mathbb{P}(i \to j \mid \text{sitting in } i \text{ at time } t) \cdot \mathbb{P}(\text{sitting in } i \text{ at time } t)$

Garpiyoruz

Which makes this probability independent on time

$= V_i(t)$

→ We are considering the structure of the web as 'Immutable' that there will be no 'deletion' or 'addition' of new links

So it becomes:

$V_j(t+1) = \sum_i \mathbb{P}(i \to j) \cdot V_i(t) \Rightarrow V_j(t+1) = \sum_i M_{ji} \cdot V_i(t)$

Now it becomes Matrix Vector Product

we can swap!

$\Rightarrow \underline{V}(t+1) = M\,\underline{V}(t)$

vector of probabilities — connection matrix 'M' — Some vector but at previous time

POWER METHOD : Straightforward tool that allows to find out one eigenvector of a square Matrix.

$A\underline{x} = \lambda\underline{x}$ → Eigenvalue (Lambda)

↑ Eigenvector ($\underline{x}$)

Eigen Pair ∀ (43. dakikada) $(\lambda, \underline{x})$

$A = [a_{ij}]_{n\times n}$ (square matrix)

- Square matrix
- General element (of A, i^{th} row j^{th} column)

$(\lambda_1, \underline{x}_1), \dots (\lambda_n, \underline{x}_n)$

λ_1: biggest eigenvalue ; $\lambda_1 \geq \lambda_i \ \forall i$

$\underline{x}_1, \dots, \underline{x}_n$ is a basis

$\underline{v}_0 = \alpha_1\underline{x}_1 + \alpha_2\underline{x}_2 + \dots + \alpha_n\underline{x}_n$ (v_0'ı weighted sum of the vector in the basis)

$\underline{v}_1 = A\underline{v}_0 = A(\alpha_1\underline{x}_1 + \alpha_2\underline{x}_2 + \dots + \alpha_n\underline{x}_n)$

$= \alpha_1 A\underline{x}_1 + \alpha_2 A\underline{x}_2 + \dots + \alpha_n A\underline{x}_n$

Can be written as

$= \alpha_1\lambda_1\underline{x}_1 + \alpha_2\lambda_2\underline{x}_2 + \dots + \alpha_n\lambda_n\underline{x}_n$

$\underline{v}_2 = A\underline{v}_1 = A(\alpha_1\lambda_1\underline{x}_1 + \dots + \alpha_n\lambda_n\underline{x}_n)$

$= \alpha_1\lambda_1^2\underline{x}_1 + \alpha_2\lambda_2^2\underline{x}_2 + \dots + \alpha_n\lambda_n^2\underline{x}_n$

⋮

$\underline{v}_k = \alpha_1\lambda_1^k\underline{x}_1 + \alpha_2\lambda_2^k\underline{x}_2 + \dots + \alpha_n\lambda_n^k\underline{x}_n$

factor-out

$\underline{v}_k = \lambda_1^k\left(\alpha_1\underline{x}_1 + \alpha_2\left(\frac{\lambda_2}{\lambda_1}\right)^k + \dots + \alpha_n\left(\frac{\lambda_n}{\lambda_1}\right)^k\underline{x}_n\right)$

$\lim_{k\to\infty} \underline{v}_k = \lim_{k\to\infty} \lambda_1^k \alpha_1 \underline{x}_1$ | Result is that as k gets bigger as $\underline{v}_k$ tends to have same direction of $\underline{x}_1$

which happens to be eigenvector associated to the first eigenvalue. (first eigenvector)

Thus, this shows 'Power Method' works! If I want to find out which is the direction associated to the first eigenvector of a matrix I can simply take the matrix, multiply it by generic vector ($\underline{v}_0$ in our case). Take the result and multiply it by 'A' and repeat it until some form of convergence takes places. The result is a vector direction identified first eigenvector $\underline{v}_0$

$$\det(A) = \sum_i a_{ij} C_{ij} = \sum_j a_{ij} C_{ij}$$

C_{ij}: MINOR

minor: Submatrix I obtained after I eliminate i^{th} row and j^{th} column

Ex:

$$M = \begin{array}{c} \begin{matrix} A & B & C & D \end{matrix} \\ \left(\begin{array}{cccc} 0 & 1/2 & 1 & 0 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 1/2 & 0 & 0 \end{array}\right) \end{array} \begin{matrix} A \\ B \\ C \\ D \end{matrix}$$

let's say $C_{12} = \begin{pmatrix} 1/3 & 0 & 1/2 \\ 1/3 & 0 & 1/2 \\ 1/3 & 0 & 0 \end{pmatrix}$

Transpose makes them

$$\det(A^T) = \sum_i a_{ij}^T C_{ij}^T = \sum a_{ji} C_{ji}$$

If we swap $i \Leftrightarrow j$

$$\det(A^T) = \sum_j a_{ij} C_{ij} = \det(A)$$

Facts:

1. $\det(A^T) = \det(A)$
2. A and A^T have same λs (eigenvalues)

$\det(A-\lambda I)=0 \iff \det(A-\lambda I)^T=0$ (should)

λ: unknown value of this equation; I: Identity Matrix (has same dimensions of A)

$\Downarrow$

$\det(A^T-\lambda I)=0$

! characteristic eqn of A and A^T are equal ∀

→ If A is row-stochastic matrix → $\sum_j a_{ij}=1$ ('i' is fixed and 'j' is running)

→ A row-wise stochastic ise 'a' elemanının i'th row fixed kalır sadece o row'a kesişik gelen column'ların toplamı ('j' is running) ve toplam 1'e eşittir.

$$A.\underline{1}=\left[\sum_j a_{ij}\,(\underline{1})_j\right]_i \rightarrow i^{th} \text{ element of vector of 1s}$$

↳ multiplying with 1 changes nothing, it is neutral ∀

$$=\left[\sum_j a_{ij}\right]_i$$

$$A.\underline{1}=[1]_i=\underline{1} \rightarrow \text{vector of 1s}$$

which means $\underline{1}$ is eigenvector of 'A'

$A.\underline{1}=1.\underline{1} \Rightarrow (1,\underline{1})$ is eigenpair of 'A'

(scaler one, vector of 1s)

We proved the 3rd fact

(3) $(1,\underline{1})$ is eigenpair of row-wise stochastic

(4) Transpose of row-wise stochastic matrix necessarily a column-wise stoch.. matrix. So, 1 is eigenvalue of column-wise stochastic Matrices.

→ One of the possible eigenvalues of the connection matrix (A) will be 1.

If A is row-wise stoch. ⇒ also A^k is row-stoch.

by induction

base: k=1 OBVIOUS

step: A^k is row-stoch

then → A^{k+1} is row-stoch matr

$$A^{k+1} = A^k \cdot A$$

a_{ij}^{k+1} → ith row and jth colm of A^{k+1} not A

generic element of A^{k+1} not A

$$a_{ij}^{k+1} = \sum_s a_{is}^k \cdot a_{sj}$$

$$\sum_j a_{ij}^{k+1} = \sum_j \sum_s a_{is}^k \cdot a_{sj}$$

$$= \sum_s a_{is}^k \sum_j a_{sj} = \sum_s a_{is}^k = 1$$

(5) If A is row-wise stochastic matrix also A^k is a row-wise stochastic ∀ 0 (∀k) for any k

→ The first eigenvalue of A column-wise stochastic matrix is 1

by absurdity initial hypothesis:

$\exists \lambda > 1$ is eigenvalue of A

→ λ is eigenvalue of A^T

$$A^T \underline{v} = \lambda \underline{v} \rightarrow (A^T)^k \underline{v} = \lambda^k \underline{v}$$

$$\sum_j (a^{T^k})_{ij} v_j = \lambda^k v_i$$

$v_{max} = \max_i v_i$ → $\sum_j (a^{T^k})_{ij} v_{max} > 6$

Set '6' as HIGH

$$\sum_j (a^{T^k})_{ij} > \frac{6}{v_{max}}$$

This sum is 1

$$1 > \frac{6}{v_{max}}$$

Contradiction

Absurdity !! 0 0

Then hypothesis become $\lambda \leq 1$ (instead $\lambda > 1$ which was absurd)

the first eigenvalue is 1

→ $M\underline{v}(0) = \underline{v}(1)$ (high enough for 'Power Method' works)

$\vdots$ } after t iteration

$\underline{v}(t) = M\underline{v}(t-1)$

$\vdots$

converges { $1.\underline{v} = M\underline{v}$ } After a suitable number of iterations we won't see any remarkable differences.

$\underline{v} = M\underline{v}$

2nd part: → The matrix will be always the same matrix whereas the vector changes each time $\forall$ Thus we need an initial vector:

1) $\underline{v}_0 = [1/n]_n$ where n: number of webpages, portion of web we are considering

2) $t = 0$ } we set variable counts current time

so, $\underline{v}_0$ is describing a particular kind of prob. dist over all the webpages we are considering. (uniform dist)

3) $\underline{v}_{t+1} = M\underline{v}_t$

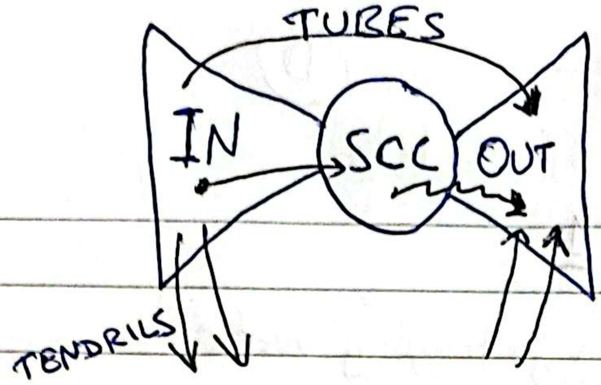
4) $t += 1$ } we increased time index

5) if $\|\underline{v}_{t+1} - \underline{v}_t\| > \varepsilon$ go to step 3, otherwise:

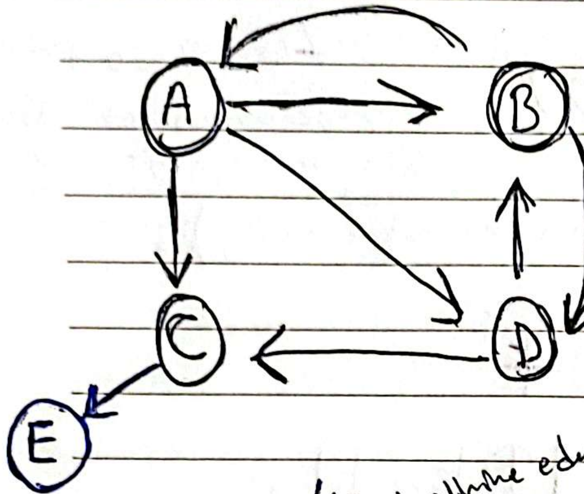
6) return $\underline{v}_{t+1}$

BOW-TIE representation of web

NCC



- Deadends (E) on the Directed graph
- Spidertraps



Deadends: (E)'nin (A) ile bağlantısı yok (B) ve (D)'ye de gitmiyor

Source: row

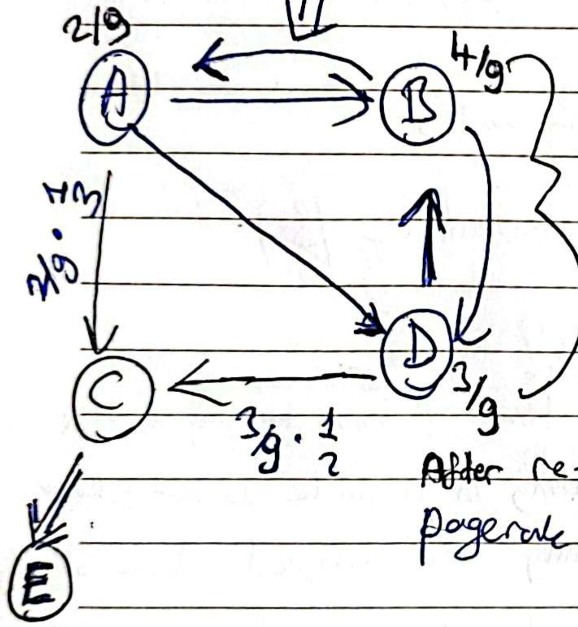
A	B	C	D	E	
0	1/2	0	0	0	A
1/3	0	0	1/2	0	B
1/3	0	0	1/2	0	C
1/3	1/2	0	0	0	D
0	0	1	0	0	E

= M

Destination: column

(E) outdegree is zero (0) → y contradictory to be a column-wise stoch

(E)'yi elimine edilirken (C)'de elimine edilmeli çünkü o da deadend kalırdı.



Deadends'ten arındırılmış hali ve (return v_{t+1}) algoritma mantıklı sonuç verebilir artık! Verdiği Pagerankleri üzerlerine yazdım.

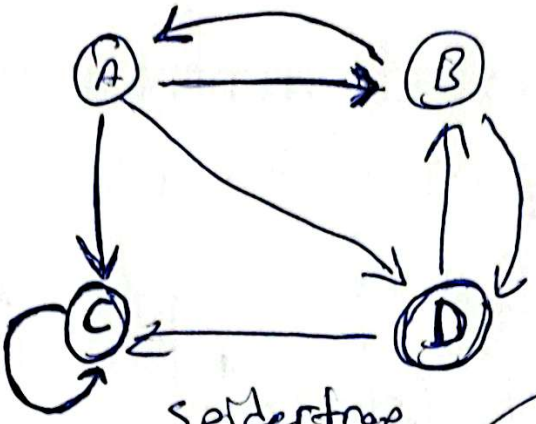
After re-introducing (C) and (E) to the graph:

pagerank of (C) $= \frac{2}{9} \cdot \frac{1}{3} + \frac{3}{9} \cdot \frac{1}{2} = \frac{13}{54}$

pagerank of (E) $= \frac{13}{54}$ since only contributor is (C) to (E)

Problem of deadends: Pagerank'lerin toplamı şuandan 1'i geçti (C) ve (E)'yi eklemeden önce bile 1 idi!

Spider traps &



Spider TRAP

TELEPORT / TAXATION

spidertrap

$$\underline{v}_{t+1} = \beta M \underline{v}_t + (1-\beta)\frac{1}{n}\cdot \underline{1}$$

$\underline{v}_0$: triggering vector

$\beta \in (0,1)$ GOOGLE USES 0.85

$\beta = P(\text{keeping on with random surfing})$

→ $\beta \approx 0.8$

eğer ola çok yakın bir değer olursa 6 aşamalı algoritmada (bir önceki sayfa) return $\underline{v}_{t+1} \Rightarrow \underline{v}_0$ gibi bir şeye output verir. 3) ve 5) arası loopa girmez!

$M\underline{v}_t \leftarrow P(\text{sitting in } i \text{ at time } t+1 \mid \text{keeping on with random surfing})$.

$\beta \leftarrow P(\text{keeping on with random surfing}) +$

$P(\text{Sitting in } i \text{ at time } t+1 \mid \underbrace{\text{Teleport}}_{\text{not keeping random surf}})$.

$\frac{1}{n} \leftarrow P(\text{Teleport})$

$\underbrace{P(\text{Teleport})}_{\text{not keeping random surfing}}$

$\Downarrow$

$(1-\beta)$ (complimentary event of β)

Why do we have also Taxation name here?

⇒ Another Possible interpretation is that each time I have to pay a tax that is some portion of the individuals who are sitting in a node do not follow the random surfing process but they are automatically re-assigned democratically in a ideal democratic society, taxes are uniformly distributed over all the population!

Why this Teleport/Taxation algorithm avoid the SpiderTraps?

→ It will not totally/completely eliminate the problem but it will somehow relieve the problem!

How Do I derive the PageRank creation process so that the result is biased towards one of those topics?

UNIVERSITÀ DEGLI STUDI DI MILANO

→ Let's say: I have fixed one topic and also I know whether or not this page says something about that topic.

topic S

I may build a vector Let's call it $\underline{S}$ that is boolean

$\underline{S}$ ⟶ (all components will be 0 or 1 accordingly it has that topic on the webpage: 1 otherwise: 0)

$$S_i = \begin{cases} 1 & \text{if page is about the topic fixed} \\ 0 & \text{, otherwise} \end{cases}$$

generic component of vector $\underline{S}$

$$\underline{V}_{t+1} = \beta M \underline{V}_t + (1-\beta)\frac{1}{|\underline{S}|} \cdot \underline{S}$$

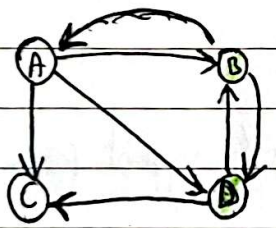
dividing $|S|$ cardinality of S instead total number of webpages 'n'

$|S|$: webpages speaks about the topic 'S'

$|\underline{S}| : \sum s_i$

The idea is that when Teleport occurs the individuals will not be uniformly assigned at random to all the available pages, rather they'll be uniformly assigned to one of the pages that speak of a particular subject. → So, If I know that utmost interest of a user is about cars, whenever that specific user gets teleported somewhere it'll be teleported to webpages speaks/related about cars!

Ex:



$S = \{B, D\}$

B and D speaks about a specific topic (topic in which my user interested in)

Topic-sensitive PageRank

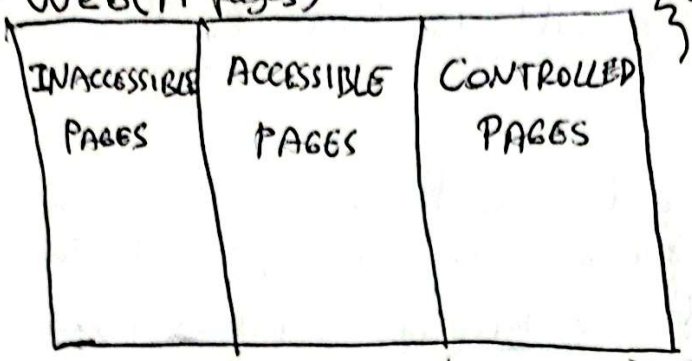
	A	B	C	D
TSPR	.26	.28	.18	.28
PR	.33	.22	.22	.22

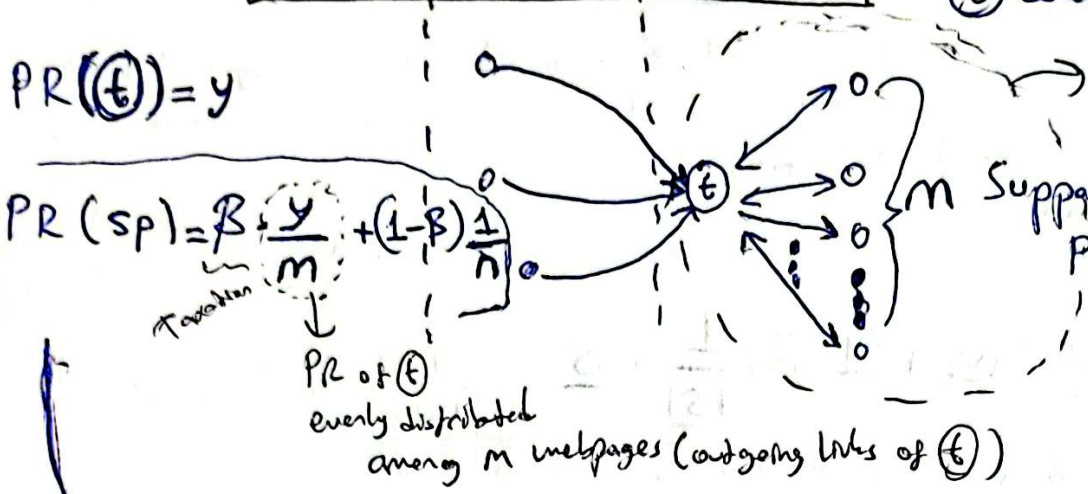
A we can say PR's dosn't (TSPR)

Web-Spammers & Artificially Increases the Page Rank of their webpages!

n: # Total webpages

Web (n pages) } we divided Venn-diagram in 3 subgroup of World Wide Web





→ Once I deducted part pertaining to TAXATION so to obtain the part that is not deducted will be evenly distributed to the all 'n' support pages. Again, because of Taxation each of 'n' support pages will recieve some pageRank of the individuals that are randomly teleported to it. (1-β) portion of the overall population is evenly distributed over all the 'n' webpages.

Let's focus again (t): PR(t) =

3 components
- x (from accessible pages)
- $(\beta \frac{y}{m} + (1-\beta)\frac{1}{n}) \cdot \beta$ ∀ support page (for each SP)
- ~~x~~ $(1-\beta)\frac{1}{n}$ } NEGLIGIBLE

SPAM MASS = $\frac{PR - TR}{PR} \simeq 1$ if TRUST RANK ↓ TR ≈ 0

So;

$$y = x + m \cdot \beta \left(\beta \frac{y}{m} + (1-\beta)\frac{1}{n}\right)$$

$$y = x + \beta^2 y + \frac{\beta(1-\beta)}{n} \cdot m$$

$$y(1-\beta^2) = x + \beta(1-\beta) \cdot \frac{m}{n}$$

$$y = \frac{x}{1-\beta^2} + \frac{\beta}{1+\beta} \cdot \frac{m}{n}$$

Suppose β = 0.8

since #n >> #m Negligible

$$PR((t)) = (3.6)x + (0.18)\frac{m}{n}$$

SPAM MADE ≈ 4 times better!