Metode za pretraživanje weba

Đurić, Antonio

Master's thesis / Diplomski rad

2022

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Science / Sveučilište u Zagrebu, Prirodoslovno-matematički fakultet**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:217:111978

SVEUČILIŠTE U ZAGREBU

PRIRODOSLOVNO–MATEMATIČKI FAKULTET

MATEMATIČKI ODSJEK

Antonio Đurić

METODE ZA PRETRAŽIVANJE WEBA

Diplomski rad

Voditelj rada:
dr. sc. Ivana Šain Glibić

Zagreb, rujan 2022.

Ovaj diplomski rad obranjen je dana _____ pred ispitnim povjerenstvom u sastavu:

1. _____, predsjednik
2. _____, član
3. _____, član

Povjerenstvo je rad ocijenilo ocjenom _____.

Potpisi članova povjerenstva:

1. _____
2. _____
3. _____

Sadržaj

Uvod

Pretraživanje weba je proces koji je mnogo kompleksniji od standardnog pretraživanja kolekcije manjih dokumenata. Zahtjevnost pretraživanja proizlazi iz ogromne količine informacija te iz mnogih drugih specifičnosti ovakve kolekcije podataka. Određene specifičnosti su: suvišni dokumenti, neispravni linkovi te dokumenti upitne kvalitete. Među ostalim web se konstantno mijenja, stalno se stvaraju nove stranice i poveznice što očito utječe na pretraživanje weba. Glavna posebnost weba koja se i koristi u definiranju metoda za njegovo pretraživanje je struktura hiperlinkova. Rad prati pregled metoda baziranih na pronalasku svojstvenog vektora za pronalaženje informacije [14] te je nadupunjen s teorijom izravno povezanom s određenim svojstvima metoda za pretraživanje weba.

U prvom poglavlju će se obraditi teorija potrebna za objašnjavanje metoda za pretraživanje weba. Prvo je objašnjena metoda potencija. Zatim, preostali dio prvog poglavlja je Perron-Frobenijusova teorija kojom kao glavni rezultat dobivamo Perron-Frobenijusov teorem. Perron-Frobenijusov teorem je ključan za raspravu jedinstvenosti rješenja prilikom korištenja metoda za pretraživanje weba. Tri metode koje će se obraditi u ovom radu, svaka u svom poglavlju, su HITS, PageRank i SALSA.

- Metoda HITS definira takozvane hubove (eng. *hubs*) i autoritete (eng. *authorities*). Pri tome autoritetima smatra one stranice prema kojima postoje poveznice s nekoliko drugih stranica, dok hubovi sadrže poveznice prema nekoliko drugih stranica. Svakoj stranici se dodjeljuje *authority score* i *hub score*. Nakon definiranja osnovnih pojmova za HITS metodu raspravlja se o konvergenciji, prednostima i manama metode te je metoda prikazana na manjem primjeru.

- Metoda PageRank svakoj stranici pridružuje *PageRank score* koji mjeri relevantnost neke stranice. Ideja je da linkovi s važnijih stranica nose veću težinu od onih s manje važnih stranica. Prikazan je Markovljev model weba koji je usko povezan s PageRank metodom. Također, obrađena je konvergencija metode, gdje bitan dio ima modifikacija takozvane Google matrice. Na kraju poglavlja je prikazana metoda na umjetnom, manjem primjeru te je poglavlje završeno s usporedbom PageRank i HITS metode.

- Metoda SALSA je kombinacija prethodne dvije navedene metode koja pokušava iskoristiti najbolja obilježja obje metode. Poglavlje koje opisuje metodu SALSA sadrži primjer metode na konkretnom primjeru.

Zadnje poglavlje sadrži konačnu usporedbu između sve tri metode. Navode se algoritmi za navedene metode za pretraživanje weba. Koristeći te algoritme promatraju se dobivene vrijednosti na većem primjeru te se promatra otpornost metoda na *spamming*.

Poglavlje 1

Metoda potencija i Perron-Frobenijusova teorija

Za početak samog rada iskazujemo sve teoreme i njihove dokaze koji će nam biti potrebni prilikom opisivanja metoda za pretraživanja weba. Odlučili smo se za ovaj pristup kako bi bilo jednostavnije za pratiti opisivanje određene metode za pretraživanje weba te ukoliko čitatelja jednostavno ne zanima potrebna teorija već opis metoda. U prvom dijelu poglavlja obrađujemo metodu potencija. Metoda potencija je algoritam koji će nam biti potreban prilikom korištenja sve tri metode za pretraživanje weba. Zatim, u drugom dijelu poglavlju govorimo u Perron-Frobenijusovoj teoremu koji je glavni rezultat ovog poglavlja. Perron-Frobenijusov teorem je ključan kada govorimo o jedinstvenosti rješenja metoda za pretraživanja weba.

1.1 Metoda potencija

Prilikom pretraživanja weba služimo se velikim matricama te posebno svojstvenim vrijednostima istih. Stoga definirajmo svojstvenu vrijednost matrice.

Definicija 1.1.1. *Neka je V vektorski prostor nad poljem $\mathbb{F}$ te neka je A linearan operator na V. Kažemo da je skalar $\lambda \in \mathbb{F}$ svojstvena vrijednost operatora A ako postoji vektor $x \in V$, $x \neq 0$ takav da vrijedi $Ax = \lambda x$.*

Napomenimo da se vektor x iz definicije za svojstvenu vrijednost λ naziva svojstvenim vektorom te je jedinstven do na množenje skalarom.

Definicija 1.1.2. *Skup svih svojstvenih vrijednosti operatora A nazivamo spektar i označavamo ga sa $\sigma(A)$.*

Glavni alat za računanje svojstvenog vektora matrice, što je ključno za pretraživanje podataka, jest algoritam zvan metoda potencija. Algoritam se pak koristi za računanje svojstvenog vektora apsolutno dominantne svojstvene vrijednost. Stoga, prije nego što opišemo algoritam definirajmo apsolutno dominantnu svojstvenu vrijednost.

Definicija 1.1.3. *Neka je A matrica sa svojstvenim vrijednostima* $\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_n$. *Kažemo da je svojstvena vrijednost* λ_1 *apsolutno dominantna ako vrijedi* $|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq \ldots \geq |\lambda_n|$.

Napomenimo kako je prva nejednakost nužno stroga nejednakost. Pokazuje se također da je od velikog značaja promatrati i najveću apsolutnu vrijednost svojstvenih vrijednosti dane matrice.

Definicija 1.1.4. *Spektralni radijus* $\rho(A)$ *matrice* $A \in \mathbb{R}^{n\times n}$ *je definiran s* $\rho(A) = \max_{\lambda\in\sigma(A)}|\lambda|$.

Prije opisivanja metode potencija definirajmo pojam dijagonalizibilne matrice.

Definicija 1.1.5. *Matrica* $A \in \mathbb{R}^{n\times n}$ *je dijagonalizabilna ako je slična nekoj dijagonalnoj matrici, tj. ako postoji regularna matrica* $S \in \mathbb{R}^{n\times n}$ *takva da je matrica* $\Lambda = S^{-1}AS$ *dijagonalna.*

Sada opišimo metodu potencija.

Algoritam 1.1.6. *Metoda potencija za računanje svojstvenog vektora za apsolutno dominantnu svojstvenu vrijednost. Neka je dana matrica A sa svojstvenim vrijednostima* $\lambda_1, \lambda_2 \ldots \lambda_n$ *numeriranim tako da vrijedi* $|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq \ldots \geq |\lambda_n|$. *Algoritam vraća svojstveni vektor apsolutno dominantne svojstvene vrijednosti.*

Algoritam 1 Metoda potencije

Require: dijagonalizibilna matrica A i početni vektor $x^{(0)}$

$y^{(0)} = \dfrac{x^{(0)}}{||x^{(0)}||}$

$k = 0$

while konvergencija nije zadovoljena **do**

$x^{(k+1)} = Ay^{(k)}$

$y^{(k+1)} = \dfrac{x^{(k+1)}}{||x^{(k+1)}||}$

$k = k + 1$

end while

return $y^{(k+1)}$

Pokažimo sada konvergenciju algoritma prema svojstvenom vektoru za apsolutno dominantnu svojstvenu vrijednost. Neka je A proizvoljna dijagonalizibilna matrica čiju svojstvenu vrijednost tražimo. Tada A možemo zapisati kao $A = S\Lambda S^{-1}$, gdje je

$$\Lambda = \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 & 0 \\ 0 & \lambda_2 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & \lambda_{n-1} & 0 \\ 0 & 0 & \cdots & 0 & \lambda_n \end{pmatrix}, \quad |\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq \ldots \geq |\lambda_n|.$$

Onda imamo da $A^k = (S\Lambda S^{-1})^k = S\Lambda^k S^{-1}$ te možemo A^k zapisati kao

$$A^k = \underbrace{S \begin{pmatrix} \lambda_1^k & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 \end{pmatrix} S^{-1}}_{B^k} + \underbrace{S \begin{pmatrix} 0 & 0 & \cdots & 0 & 0 \\ 0 & \lambda_2^k & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & \lambda_{n-1}^k & 0 \\ 0 & 0 & \cdots & 0 & \lambda_n^k \end{pmatrix} S^{-1}}_{C^k}$$

pri čemu se A^k, kako k raste, bliži matrici ranga jedan:

$$\frac{\left\|A^k - B^k\right\|_2}{\left\|A^k\right\|_2} = \frac{\left\|C^k\right\|_2}{|\lambda_1|^k} \leq \|S\|_2 \left\|S^{-1}\right\|_2 \left|\frac{\lambda_2}{\lambda_1}\right|^k \to 0.$$

Dani izraz, uzimanjem limesa kada $k \to \infty$, teži u nulu jer $\lambda_1 > \lambda_2$. Dakle, A^k je blizu matricu ranga jedan te bi onda $A^k x$ trebao dati dobru informaciju o svojstvenom vektoru matrice A.

Napomena 1.1.7. *U prethodnoj nejednakosti korisitili smo jednakost* $\|A\|_2 = \sqrt{\rho(A^*A)}$. *Neka su* $\mu_1 \geq \mu_2 \geq \cdots \geq \mu_n$ *svojstvene vrijednosti od* A^*A. *Tada po definicji norme imamo:*

$$\|A\|_2^2 = \max_{\|x\|_2=1} \|Ax\|_2^2 = \max_{\|x\|_2=1} \langle Ax, Ax\rangle = \max_{\|x\|_2=1} \langle A^*Ax, x\rangle = \mu_1.$$

Konačno, dobivamo traženu nejednakost $\|A\|_2 = \sqrt{\mu_1} = \sqrt{\rho(A^*A)}$.

Iako metoda potencija uvijek konvergira prema nekom vektoru, taj vektor ne mora nužno biti jedinstven. Stoga trebamo teoriju numeričke analize kojom bi osigurali jedinstvenost rješenja. Već vidimo koliko su svojstvene vrijednosti i svojstveni vektori od iznimnog značaja u teoriji te i u praktičnim numeričkim algoritmima.

1.2 Perron-Frobenijusov teorem

U ovom dijelu poglavlja dokazujemo Perron-Frobenijusov teorem. Pristup dokazivanja teorema je sličan onima napravljenim u skripti [9] i knjizi [17] te su korištene propozicije i teoremi iz navedenih izvora.

Jedan od ključnih teorema potreban za dokazivanje jedinstvenosti rješenja glasi:

Teorem 1.2.1. *Za proizvoljnu matricu $A \in \mathbb{R}^{n\times n}$ vrijedi:*

$$\lim_{k\to\infty} A^k = 0 \iff \rho(A) < 1.$$

Dokaz. Pretpostavimo da je $\lim_{k\to\infty} A^k = 0$.
Neka je λ svojstvena vrijednost matrice A te x svojstveni vektor za λ. Jer vrijedi $A^k x = \lambda^k x$ imamo:

$$\begin{aligned} 0 &= \left(\lim_{k\to\infty} A^k\right) x \\ &= \lim_{k\to\infty} \left(A^k x\right) \\ &= \lim_{k\to\infty} \left(\lambda^k x\right) \\ &= x \cdot \lim_{k\to\infty} \lambda^k. \end{aligned}$$

Po definciji svojstvenog vektora znamo da $x \neq 0$. Stoga slijedi da $\lim_{k\to\infty} \lambda^k = 0$. No λ je skalar te stoga mora biti $\lambda < 1$. Ovo vrijedi za proizvoljnu svojstvenu vrijednost λ te onda vrijedi $\rho(A) < 1$.

Pretpostavimo sada da vrijedi $\rho(A) < 1$. Koristeći teorem o Jordanovoj normalnoj formi [18] znamo da za svaku matricu A postoji matrica permutacije P i blok dijagonalna matrica J takve da $A = PJP^{-1}$ gdje je

$$J = \begin{bmatrix} J_{m_1}(\lambda_1) & 0 & \cdots & 0 & 0 \\ 0 & J_{m_2}(\lambda_2) & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & J_{m_{s-1}}(\lambda_{s-1}) & 0 \\ 0 & 0 & \cdots & 0 & J_{m_s}(\lambda_s) \end{bmatrix}$$

i gdje je

$$J_{m_i}(\lambda_i) = \begin{bmatrix} \lambda_i & 1 & 0 & \cdots & 0 \\ 0 & \lambda_i & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_i & 1 \\ 0 & 0 & \cdots & 0 & \lambda_i \end{bmatrix} \in \mathbb{R}^{m_i\times m_i} \quad , 1 \leq i \leq s.$$

Primijetimo da vrijedi

$$A^k = \left(PJP^{-1}\right)^k = PJ^kP^{-1}.$$

Jer je J blok dijagonalna imamo:

$$J^k = \begin{bmatrix} J^k_{m_1}(\lambda_1) & 0 & \cdots & 0 & 0 \\ 0 & J^k_{m_2}(\lambda_2) & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & J^k_{m_{s-1}}(\lambda_{s-1}) & 0 \\ 0 & 0 & \cdots & 0 & J^k_{m_s}(\lambda_s) \end{bmatrix}.$$

Također zaključimo da su potencije Jordanovih blokova oblika

$$J^k_{m_i}(\lambda_i) = \begin{bmatrix} \lambda_i^k & \binom{k}{1}\lambda_i^{k-1} & \binom{k}{2}\lambda_i^{k-2} & \cdots & \binom{k}{m_i-1}\lambda_i^{k-m_i+1} \\ 0 & \lambda_i^k & \binom{k}{1}\lambda_i^{k-1} & \cdots & \binom{k}{m_i-2}\lambda_i^{k-m_i+2} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_i^k & \binom{k}{1}\lambda_i^{k-1} \\ 0 & 0 & \cdots & 0 & \lambda_i^k \end{bmatrix}.$$

Iskoristimo konačno pretpostavku $\rho(A) < 1$. Tada slijedi $|\lambda_i| < 1$ za svaki $1 \leq i \leq s$. Stoga za svaki i vrijedi:

$$\lim_{k\to\infty} J^k_{m_i} = 0$$

te onda $\lim_{k\to\infty} J^k = 0$ te konačno

$$\lim_{k\to\infty} A^k = \lim_{k\to\infty} PJ^kP^{-1} = P\underbrace{\left(\lim_{k\to\infty} J^k\right)}_{0}P^{-1} = 0.$$

□

Također nam je potreban sljedeći teorem.

Teorem 1.2.2. *Neka je $\|\cdot\|$ proizvoljna matrična forma na $\mathbb{R}^{n\times n}$. Tada je za svaku matricu $A \in \mathbb{R}^{n\times n}$*

$$\rho(A) = \lim_{k\to\infty} \|A^k\|^{1/k}.$$

Dokaz. Pokažimo da za proizvoljnu matričnu normu $\|\cdot\|$ na $\mathbb{R}^{n\times n}$ i za proizvoljnu matricu $A \in \mathbb{R}^{n\times n}$ vrijedi $\rho(A) < \|A\|$.

Neka je λ svojstvena vrijednost matrice A te x pripadni svojstveni vektor. Definirajmo matricu $X \neq 0$ tako da je svaki stupac matrice upravo svojstveni vektor x.
Tada očito vrijedi $AX = \lambda X$. Uzimanjem proizvoljne norme imamo

$$|\lambda| \, ||X|| = ||AX|| \leq ||A|| ||X||.$$

Vidimo da za proizvoljnu svojstvenu vrijednost λ vrijedi $|\lambda| \leq ||A||$ te onda i $\rho(A) \leq ||A||$.

Koristeći dokazanu tvrdnju imamo $\rho(A)^k = \rho\left(A^k\right) \leq ||A^k||$ gdje jednakost slijedi iz činjenice da ako je λ svojstvena vrijednost matrice A, tada je λ^k svojstvena vrijednost matrice A^k.

Tada iz ove relacije imamo $\rho(A) \leq ||A^k||^{1/k}$. Neka je $\epsilon > 0$ i definirajmo $B = \dfrac{A}{\rho(A) + \epsilon}$. Jer $\rho(B) = \dfrac{\rho(A)}{\rho(A) + \epsilon} < 1$ iz teorema 1.2.1 vrijedi da $\lim_{k \to \infty} B^k = 0$. Tada postoji nekakav indeks k_0 takav da je za sve $k \geq k_0$ $||B^k|| < 1$. Tada po definiciji matrice B slijedi da $||A^k|| < (\rho(A) + \epsilon)^k$.

Konačno imamo:

$$\begin{aligned} \rho(A)^k \leq ||A^k|| < (\rho(A) + \epsilon)^k \quad & k \geq k_0 \\ \rho(A) \leq ||A^k||^{1/k} < \rho(A) + \epsilon \quad & k \geq k_0. \end{aligned}$$

Tvrdnja vrijedi za proizvoljan $\epsilon > 0$. Stoga uzimnanjem limesa sa svake strane dobivamo traženu tvrdnju. □

Sada koristeći ovaj teorem možemo dokazati sljedeću tvrdnju.

Teorem 1.2.3. *Neka su $A, B \in \mathbb{R}^{n \times n}$. Vrijedi $\rho(A) \leq \rho(|A|)$. Također, ako vrijedi $|A| \leq B$, onda je $\rho(|A|) \leq \rho(B)$, gdje $|A|$ označava matricu dobivenu uzimanjem apsolutne vrijednosti po elementima matrice A.*

Dokaz. Prvo primijetimo da vrijedi $\left|A^k\right| \leq |A|^k$. Sada uz pomoć ove nejednakosti te teorema 1.2.2 imamo:

$$\begin{aligned} & \left\|A^k\right\|_\infty \leq \left\| \left|A^k\right| \right\|_\infty \leq \left\| |A|^k \right\|_\infty \leq \left\|B^k\right\|_\infty \\ \implies & \left\|A^k\right\|_\infty^{1/k} \leq \left\| |A|^k \right\|_\infty^{1/k} \leq \left\|B^k\right\|_\infty^{1/k} \\ \implies & \lim_{k \to \infty} \left\|A^k\right\|_\infty^{1/k} \leq \lim_{k \to \infty} \left\| |A|^k \right\|_\infty^{1/k} \leq \lim_{k \to \infty} \left\|B^k\right\|_\infty^{1/k} \\ \implies & \rho(A) \leq \rho(|A|) \leq \rho(B). \end{aligned}$$

□

U svim metodama za pretraživanje weba koristimo matrice sa realnim nenegativnim ili elementima. Za nenegativne matrice razvijena je tzv. Perron-Frobenijusova teorija kojom ćemo moći na određen način prisiliti jedinstvenost rješenja.
Prije iskazivanja samog Perron-Frobenijusovog teorema dokazat ćemo nekoliko pomoćnih rezultata.

Definicija 1.2.4. *Neka je $A \in \mathbb{R}^{n\times n}$ proizvoljna matrica. Kažemo da je A nenegativna ako $a_{ij} \geq 0$ za sve $0 \leq i, j \leq n$. Oznaka je $A \geq 0$.*

Definicija 1.2.5. *Neka je $A \in \mathbb{R}^{n\times n}$ proizvoljna matrica. Kažemo da je A pozitivna ako $a_{i,} > 0$ za sve $0 \leq i, j \leq n$. Oznaka je $A > 0$.*

Definicija 1.2.6. *Neka su $A, B \in \mathbb{R}^{n\times n}$ proizvoljne matrice. Kažemo da je $A \geq B$ ako $a_{ij} \geq b_{ij}$ za sve $0 \leq i, j \leq n$.*

Propozicija 1.2.7. *Neka je $A \in \mathbb{R}^{n\times n}$ nenegativna matrica. Tada vrijedi:*

$$\min_{i\leq 1\leq n} \sum_{j=1}^{n} a_{ij} \leq \rho(A) \leq \max_{i\leq 1\leq n} \sum_{j=1}^{n} a_{ij}.$$

Nadalje, za svaki vektor $x > 0$ je

$$\min_{i\leq 1\leq n} \frac{1}{x_i} \sum_{j=1}^{n} a_{ij}x_j \leq \rho(A) \leq \max_{i\leq 1\leq n} \frac{1}{x_i} \sum_{j=1}^{n} a_{ij}x_j.$$

Ako je za neki $x > 0$ i $\alpha, \beta \geq 0$ zadovoljeno $\alpha x \leq Ax \leq \beta x$, onda je $\alpha \leq \rho(A) \leq \beta$.

Dokaz. Na početku teorema 1.2.2 pokazali smo da za proizvoljnu matričnu normu $\|\cdot\|$ na $\mathbb{R}^{n\times n}$ i za proizvoljnu matricu $A \in \mathbb{R}^{n\times n}$ vrijedi $\rho(A) < \|A\|$.
Stoga imamo gornju ogradu $\rho(A) \leq \|A\|_\infty = \max_{i\leq 1\leq n} \sum_{j=1}^{n} a_{ij}$.

Označimo $\gamma = \min_{i\leq 1\leq n} \sum_{j=1}^{n} a_{ij}$. Definirajmo matricu B na sljedeći način. Ako je $\gamma = 0$ onda $B = 0$, inače $b_{ij} = a_{ij}\gamma / \sum_{k=1}^{n} a_{ik}$ za sve i, j.

Možemo vidjeti $0 \leq B \leq A$ jer $\gamma \leq \sum_{k=1}^{n} a_{ik}$ za bilo koji i. Po teoremu 1.2.3 vrijedi $\rho(B) \leq \rho(A)$.

Nadalje, vidimo da za svaki redak i vrijedi $\sum_{j=1}^{n} b_{ij} = \gamma$. No jer su sume po svim retcima jednake, tada je vektor 1 svojstveni vektor matrice B pridružen svojstvenoj vrijednosti $\gamma = \|B\|_\infty$ te vrijedi $\gamma = \|B\|_\infty \leq \rho(B) \leq \|B\|_\infty$.
Konačno, imamo $\gamma = \rho(B) \leq \rho(A)$ i time smo dokazali prvu nejednakost.

Za drugu nejednakost napravimo dijagonalnu matricu $S = \operatorname{diag}(x_1, \ldots, x_n)$, $x_i \in \mathbb{R}$. Sada, pomoću upravo dokazane tvrdnje za matricu $S^{-1}AS$ i jednakosti $\rho(S^{-1}AS) = \rho(A)$, slijedi tražena tvrdnja.

Konačno, za zadnju tvrdnju pretpostavimo da za neki $x > 0$ i $\alpha \geq 0$ zadovoljeno $\alpha x \leq Ax$. Dijeljenjem sa x i koristeći prethodno dokazanu nejednakost imamo:

$$\alpha \leq \min_{i \leq 1 \leq n} \frac{1}{x_i} \sum_{j=1}^{n} a_{ij} x_j \leq \rho(A).$$

Na isti način dobivamo drugu nejednakost. □

Napomenimo da za zadnju tvrdnju prethodnog teorema možemo dobiti strogu nejednakost tako da za neki $\alpha \geq 0$ takav da $\alpha x < Ax$ možemo pronaći $\tilde{\alpha} > \alpha$ takav da vrijedi $\tilde{\alpha} x \leq Ax$, što daje $\alpha < \tilde{\alpha} \leq \rho(A)$.
Koristeći ovaj teorem dobivamo sljedeći posljedicu.

Korolar 1.2.8. *Neka je $A \in \mathbb{R}^{n \times n}$ nenegativna matrica. Ako A ima pozitivan svojstveni vektor $v > 0$ onda je pripadna svojstvena vrijednost jednaka $\rho(A)$.*

Dokaz. Po definiciji svojstvene vrijednost imamo jednakost $Av = \lambda v$ i po pretpostavci imamo $v > 0$. Iz toga slijedi nenegativnost i realnost svojstvene vrijednosti. Zatim iz nejednakosti $\lambda v \leq Av \leq \lambda v$ pomoću propozicije 1.2.7 imamo $\lambda \leq \rho(A) \leq \lambda$. Odnosno $\lambda = \rho(A)$. □

Sada možemo dokazati Perronov teorem za pozitivne matrice kojeg ćemo kasnije proširiti na nenegativne i ireducibilne matrice koji nam izravno osigurava jedinstvenost rješenja prilikom korištenja metoda za pretraživanje weba.

Definicija 1.2.9. *Kažemo da je kvadratna matrica $A \in \mathbb{R}^{n \times n}, n \geq 2$ reducibilna ako postoji matrica permutacije P i $m \in \{1, \ldots, n-1\}$ takvi da*

$$P^T A P = \begin{pmatrix} X & Y \\ 0 & Z \end{pmatrix}, \quad X \in \mathbb{R}^{m \times m}.$$

Kažemo da je matrica A ireducibilna ako nije reducibilna.

Teorem 1.2.10. *Neka je $A \in \mathbb{R}^{n \times n}$ pozitivna matrica. Tada $\rho(A) > 0$ i $\rho(A)$ je svojstvena vrijednost od A, pri čemu se pripadni svojstveni vektor v može odabrati sa pozitivnim koeficijentima, $Av = \rho(A)v, v > 0$.*

Dokaz. Iz propozicije 1.2.7 zbog pozitivnosti matrice slijedi $\rho(A) > 0$. Neka je $\lambda \in \sigma(A)$ svojstvena vrijednost za koju je $\rho(A) = |\lambda|$ te neka je $x \neq 0$ pripadni svojstveni vektor za λ, $Ax = \lambda x$. Sada uzimanjem apsolutne vrijednosti imamo:

$$\rho(A)|x| = |\lambda||x| = |\lambda x| = |Ax| \leq |A||x| = A|x|\,.$$

Definirajmo sa $y = A|x| - \rho(A)|x|$. Po prethodnoj relaciji očito $y \geq 0$. Promotrimo dva slučaja.

1. $y = 0$
   Onda po definciji y slijedi $A|x| = \rho(A)|x|$. Odnosno, $\rho(A)$ je svojstvena vrijednost s pripadnim svojstvenim vektorom $v = |x| \neq 0$.
   Također imamo:
   $$\rho(A)v = Av \implies v = \rho(A)^{-1}Av$$
   te stoga vrijedi $v > 0$.

2. $y > 0$
   Pokažimo da je ova situacija nemoguća. U ovom slučaju zbog pozitivnosti matrice A vrijedi da je $Ay > 0$. No tada sa oznakom $z = A|x| > 0$ imamo
   $$Ay > 0 \implies Az - \rho(A)z > 0,$$
   odnosno $Az > \rho(A)z$.
   Međutim, po zadnjem dijelu propozicije 1.2.7 iz $Az > \rho(A)z$ slijedi $\rho(A) > \rho(A)$ te možemo zaključiti kako nije moguće da je $y > 0$.

□

Dokažimo sada isti teorem za nenegativne matrice.

Teorem 1.2.11. *Neka je $A \in \mathbb{R}^{n\times n}$ nenegativna matrica. Tada je $\rho(A)$ svojstvena vrijednost od A, pri čemu se pripadni svojstveni vektor v može odabrati sa pozitivnim koeficijentima, $Av = \rho(A)v, v > 0$.*

Dokaz. Ideja dokaza je prikazati nenegativni matricu kao limes pozitivnih matrica. Stoga, uzmimo strogo padajući niz pozitivnih brojeva (ϵ_k) takav da $\lim_{k\to\infty} \epsilon_k = 0$.
Definirajmo sada niz pozitivnih matrica izrazom

$$A(\epsilon_k) = A + \epsilon_k \mathbb{1} \cdot \mathbb{1}^{\mathrm{T}},$$

gdje je $\mathbb{1} = (1, \ldots, 1)$. Po prethodnom teoremu za svaki ϵ_k možemo pisati

$$A(\epsilon_k)v(\epsilon_k) = \rho\left((A(\epsilon_k)\right)v(\epsilon_k), \tag{1.1}$$

gdje je $v(\epsilon_k) > 0$ i $\left\|v(\epsilon_k)\right\|_1 = 1$.

Niz svojstvenih vektora je sadržan u kompaktnom skupu pa stoga niz ima konvergentan podniz. Neka je $\lim_{j\to\infty} v(\epsilon_{k_j}) = v$. Očito vrijedi $v > 0$ i $\|v\|_1 = 1$.

Nadalje, za odgovarajući podniz (ϵ_{k_j}) vrijedi $\epsilon_{k_j} > \epsilon_{k_{j+1}}$ za $j = 1, 2, \ldots$ te $\lim_{j\to\infty} \epsilon_{k_j} = 0$. Kako je $A\left(\epsilon_{k_j}\right) > A\left(\epsilon_{k_{j+1}}\right)$ po definiciji $A(\epsilon_k)$, pomoću teorema 1.2.3 slijedi da je $\rho\left(A\left(\epsilon_{k_j}\right)\right) \geq \rho\left(A\left(\epsilon_{k_{j+1}}\right)\right) \geq \rho(A)$. Imamo padajući, odozdo omeđeni niz, stoga postoji limes $L = \lim_{j\to\infty} \rho\left(A\left(\epsilon_{k_j}\right)\right)$ i vrijedi $L \geq \rho(A)$.

Konačno, ako u izrazu 1.1 uzmemo podniz $j \to k_j$ te promotrimo limes izraza kada j teži prema beskonačno dobivamo:

$$\begin{aligned}\lim_{j\to\infty} A(\epsilon_{k_j})v(\epsilon_{k_j}) &= \lim_{j\to\infty} \rho\left((A(\epsilon_{k_j})\right) v(\epsilon_{k_j}) \\ Av &= Lv.\end{aligned}$$

Iz ovog izraza po propoziciji 1.2.7 slijedi $L \geq \rho(A)$, što konačno daje $L = \rho(A)$. □

Prije iskaza, te i dokaza, samog Perron-Frobenijusevog teorema iskažimo prvo dvije tehničke propozicije potrebne za dokaz samog teorema.

Propozicija 1.2.12. *Ako je B proizvoljna glavna podmatrica ireducibilne nenegativne matrice A onda je $\rho(B) < \rho(A)$.*

Dokaz. Definirajmo $\tilde{A} = \begin{pmatrix} B & 0 \\ 0 & 0 \end{pmatrix}$. Jer je A ireducibilna slijedi $A > \tilde{A}$. Sada iz teorema 1.2.3 slijedi $\rho\left(\tilde{A}\right) < \rho\left(A\right)$ (primijetimo u dokazu teorema 1.2.3 da možemo koristiti strogu nejednakost). Također, očito vrijedi $\rho\left(\tilde{A}\right) > \rho(B)$ po konstrukciji $\tilde{A}$. Koristeći obje nejednakosti dobivamo traženu tvrdnju. □

Propozicija 1.2.13. *Neka je $\chi_A(\lambda) = \det(\lambda I - A)$ karakteristični polinom realne $n \times n$ matrice A. Tada je*

$$\frac{\mathrm{d}}{\mathrm{d}\lambda}\chi_A(\lambda) = \sum_{i=1}^{n} \det\left(\lambda I_{n-1} - A_{]i,i[}\right),$$

gdje $A_{]i,i[}$ označava $(n-1)\times(n-1)$ podmatricu od A, dobivenu izbacivanjem i-tog stupca i i-tog retka matrice A.

Konačno dokažimo Perron-Frobenijusov teorem.

Teorem 1.2.14. *Neka je $A \in \mathbb{R}^{n\times n}$ nenegativna i ireducibilna matrica. Tada $\rho(A) > 0$ i $\rho(A)$ je svojstvena vrijednost od A algebarske kratnosti 1, pri čemu se pripadni svojstveni vektor v može odabrati sa pozitivnim koeficijentima, $Av = \rho(A)v$, $v > 0$.*

Dokaz. U teoremu 1.2.11 smo pokazali za nenegativnu matricu da je $\rho(A)$ svojstvena vrijednost od A s pripadnim pozitivnim svojstvenim vektorom v. Nadalje, matrica A je ireducibilna i stoga ne može imati nul redak pa zbog 1.2.7 vrijedi $\rho(A) > 0$.

Preostaje nam dokazati da je svojstvena vrijednost $\rho(A)$ algebarske kratnosti 1. Neka je $t \in \mathbb{R}$ bilo koji broj takav da $t \geq \rho(A)$. Tada je po propoziciji 1.2.12 nužno $\det\left(tI_{n-1} - A_{]i,i[}\right) \neq 0$. Kako je $\lim_{t\to\infty} \det\left(tI_{n-1} - A_{]i,i[}\right) = \infty$, mora biti $\det\left(\rho(A)I_{n-1} - A_{]i,i[}\right) > 0$, pa je $\chi'_A(\rho(A)) > 0$ po propoziciji 1.2.13. Dakle $\rho(A)$ nije nultočka derivacije karakterističnog polinoma, onda je $\rho(A)$ jednostruka nultočka karakterističnog polinoma, odnosno algebarske je kratnosti 1. □

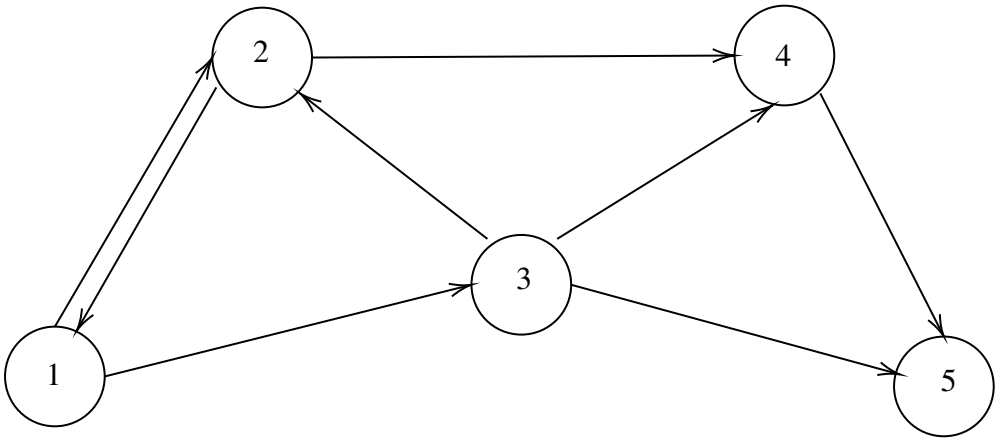
Napomenimo kako ovo nije u cijelosti Perron-Frobenijusov teorem, već je dio teorema koji je nama potreban za osigurati jedinstvenost rješenja metoda za pretraživanje weba. Više informacija o Perron-Frobenijusovoj teoriji može se pronaći u [17].

Poglavlje 2

HITS

U ovom poglavlju opisujemo HITS (Hypertext Induced Topic Search) metodu za pretraživanje weba. HITS metodu je razvio Jon Kleinberg 1997. godine.

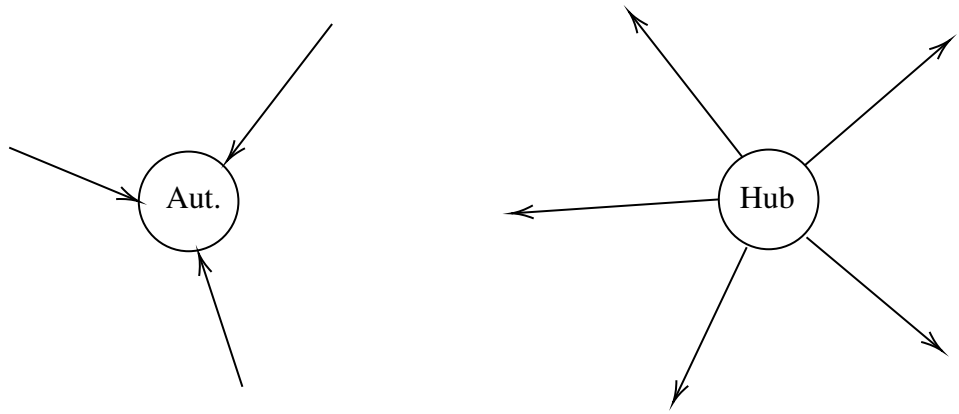
Ideja je da zamišljamo stranicu na webu kao vrh u velikom grafu. Usmjereni bridovi u tom grafu predstavljaju poveznice s jedne stranice na drugu. Prikaz takve strukture možemo vidjeti na slici 2.1.



Slika 2.1: Prikaz strukture weba

U HITS metodi za pretraživanje weba razlikujemo dvije vrste vrhova. Autoritet (eng. *authority*) je dokument/stranica na webu koja ima nekoliko ulaznih linkova (*inlinks*), hub (eng. *hub*) nazivamo dokument/stranicu na webu koja ima nekoliko izlaznih linkova (*outlinks*). Primjer autoriteta i huba možemo vidjeti na slici 2.2.

Glavna ideja HITS metode je da dobri hubovi pokazuju na dobre autoritete te da su prema dobrim autoritetima usmjereni dobri hubovi. Probajmo sada to na neki način eksplicitno napisati.



Slika 2.2: Autoritet i hub vrhovi

Neka je i neka stranica na webu. Tada označimo s x_i vrijednost autoriteta te s y_i hub vrijednost. Također, označimo s E skup svih usmjerenih bridova u grafu weba gdje e_{ij} predstavlja usmjerenih brid iz vrha i prema vrhu j. Pretpostavimo sada da imamo zadanu početnu vrijednost autoriteta $x_i^{(0)}$ i početnu hub vrijednost $y_i^{(0)}$. HITS metoda se zasniva na dorađivanju početnih vrijednosti sljedećim formulama:

$$x_i^{(k)} = \sum_{j:e_{ji}\in E} y_j^{(k-1)} \quad \text{i} \quad y_i^{(k)} = \sum_{j:e_{ij}\in E} x_j^{(k)} \qquad k = 1, 2, \ldots \tag{2.1}$$

Riječima, vrijednost autoriteta x_i u k-tom koraku je suma hub vrijednosti u $(k-1)$-om koraku svih hub čvorova koji pokazuju na čvor autoriteta i.
Slično, hub vrijednost y_i u k-tom koraku je suma vrijednosti u k-tom koraku svih čvorova autoriteta na koje pokazuje hub čvor j.

Jednadžbe 2.1 možemo elegantnije zapisati koristeći matricu susjedstva $\mathbf{L}$ usmjerenog web grafa. Definiramo:

$$\mathbf{L}_{ij} = \begin{cases} 1, & \text{ako postoji brid iz vrha } i \text{ prema vrhu } j \\ 0, & \text{inače} \end{cases}. \tag{2.2}$$

Koristeći definiciju matrice $\mathbf{L}$ dobivamo sljedeći prikaz jednadžbi 2.1:

$$\mathbf{x}^{(k)} = \mathbf{L}^T\mathbf{y}^{(k-1)} \quad \text{i} \quad \mathbf{y}^{(k)} = \mathbf{L}\mathbf{x}^{(k)}.$$

Međusobnim uvrštavanjem prethodnih jednadžbi dobivamo sljedeće jednakosti

$$\mathbf{x}^{(k)} = \mathbf{L}^T\mathbf{L}\mathbf{x}^{(k-1)} \quad \text{i} \quad \mathbf{y}^{(k)} = \mathbf{L}\mathbf{L}^T\mathbf{y}^{(k-1)}.$$

Sada vektore $\mathbf{x}$ i $\mathbf{y}$ možemo pronaći koristeći metodu potencija 1.1.6. Kako matricu $\mathbf{L}^T\mathbf{L}$ koristimo za dobivanje vrijednosti autoriteta matricu nazivamo matricom autoriteta. Također, matricu $\mathbf{L}\mathbf{L}^T$ koristimo za određivanje hub vrijednosti pa ju nazivamo hub matricom.

2.1 Konvergencija HITS metode

Za početak uvedimo oznake za prije navedene umnoške matrica.

Neka je $A = \mathbf{L}^T\mathbf{L}$, $B = \mathbf{L}\mathbf{L}^T$ te neka n označava broj redaka, odnosno stupaca matrice $\mathbf{L}$.

Nadalje, podsjetimo se metode potencija 1.1.6. Kako bi algoritam konvergirao nužno je da je matrica dijagonalizibilna te za njezine stvojstvene vrijednosti $\lambda_1, \lambda_2, \ldots, \lambda_n$ vrijedi $|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq \ldots \geq |\lambda_n|$.

Primijetimo da su matrice A i B simetrične. Stoga, svojstvene vrijednosti matrica A i B su realne.

Dodatno, za matrice A i B vrijedi $v^T Av \geq 0$, odnosno $v^T Bv \geq 0$ za sve vektore $v \in \mathbb{R}^n$. To možemo vidjeti na sljedeći način:

$$\begin{aligned} v^T Av &= v^T \mathbf{L}^T \mathbf{L} v \\ &= (\mathbf{L}v)^T (\mathbf{L}v) \\ &= (\mathbf{L}v) \cdot (\mathbf{L}v) \geq 0, \end{aligned}$$

$$\begin{aligned} v^T Bv &= v^T \mathbf{L}\mathbf{L}^T v \\ &= \left(\mathbf{L}^T v\right)^T \left(\mathbf{L}^T v\right) \\ &= \left(\mathbf{L}^T v\right) \cdot \left(\mathbf{L}^T v\right) \geq 0. \end{aligned}$$

Ovaj račun nam daje motivaciju za sljedeću definiciju.

Definicija 2.1.1. *Neka je $S \in \mathbb{R}^{n\times n}$ proizvoljna simetrična matrica. Kažemo da je S pozitivno semidefinitna ako je $x^T S\, x \geq 0$ za svaki vektor $x \in \mathbb{R}^n$.*

Pokazali smo prethodno da su matrice A i B pozitivno semidefinitne. Štoviše, pokazali smo da bilo koja simetrična matrica koju možemo zapisati kao produkt matrica $X^T X$ ili XX^T je pozitivno semidefinitna.

Propozicija 2.1.2. *Svojstvene vrijednosti proizvoljne pozitivne semidefinitne matrice su nenegativne.*

Dokaz. Neka je $S \in \mathbb{R}^{n\times n}$ proizvoljna pozitivna semidefinitna matrica. Tada po definiciji vrijedi $x^T S\, x \geq 0$ za svaki $x \in \mathbb{R}^n$. Neka je λ svojstvena vrijednost matrice S.

Tada po definiciji svojstvene vrijednosti postoji netrivijalan svojstveni vektor $v \in \mathbb{R}^n$ takav da $S\, v = \lambda v$. Zatim, iz pozitivne semidefinitnosti matrice posebno za vektor v imamo:

$$0 \leq v^T S\, v = v^T \lambda v = \lambda v^T v.$$

Znamo da je $v^T v$ sigurno strogo veće od nule. Iz čega slijedi $\lambda \geq 0$. □

Pokazali smo iz svojstava matrica A i B da su svojstvene vrijednosti $\{\lambda_1, \ldots, \lambda_n\}$ realne i strogo negativne. Bez smanjenja općenitosti stavimo $\lambda_1 > \lambda_2 > \cdots > \lambda_n > 0$.

Prateći dokaz konvergencije algoritma 1.1.6 prema svojstvenom vektoru za apsolutnu dominanantnu svojstvenu vrijednost možemo vidjeti kako je ovo dovoljno za konvergenciju. No, također možemo primijetiti da u slučaju ponavljajuće svojstvene vrijednosti λ_1 algoritam ne garantira jedinstveno rješenje, odnosno različiti početni vektori metode potencija mogu generirati različita rješenja.

Primjer 2.1.3. *Neka je* $\mathbf{L} = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$. *Tada je* $\mathbf{L}^T\mathbf{L} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$.

Iz ovog odmah vidimo da su svojstvene vrijednosti matrice $\mathbf{L}^T\mathbf{L}$ *jednake* $\lambda_1 = 1$ *te* $\lambda_2 = 0$. *Algebarska kratnost svojstvene vrijednost* $\lambda_1 = 1$ *je dva.*

Sada izračunajmo vrijednosti vektora autoriteta koristeći metodu potencija. Također, napemenimo kako možemo koristiti proizvoljnu normu za normiranje vekotra. U ovom slučaju koristimo 1*-normu. Prvo, uzmimo vektor* $\mathbf{x}^{(0)} = (0 \quad 1/3 \quad 2/3)^T$ *za početni vektor. Metodom potencije dobijemo vektor* $\mathbf{x}^{(\infty)} = (0 \quad 1 \quad 0)^T$.

Zatim, uzmimo vektor $\mathbf{x}^{(0)} = (1/2 \quad 1/4 \quad 1/4)^T$ *za početni. Metodom potencije dobijemo vektor* $\mathbf{x}^{(\infty)} = (2/3 \quad 1/3 \quad 0)^T$.

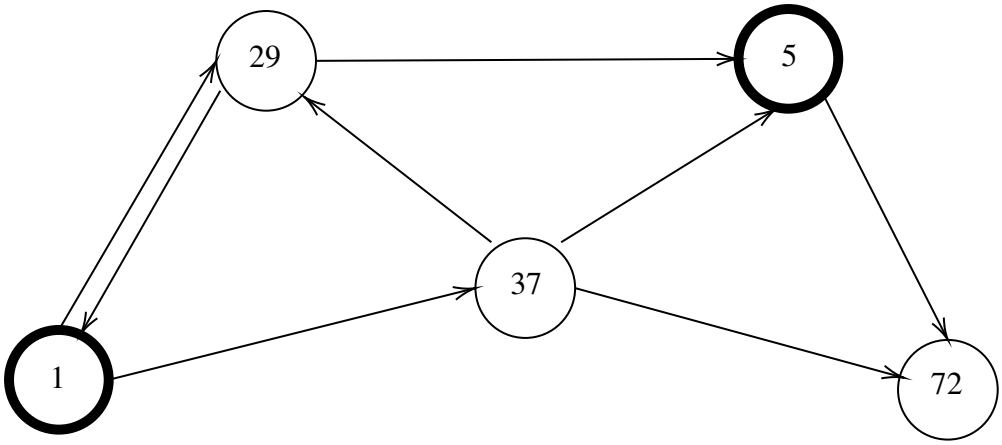
Kao što vidimo u prethodnom primjeru za dva različita početna vektora metoda potencija je dala dva različita vektora. Problem jedinstvenosti upravo leži u reducibilnosti matrice. Kada bi matrica bila ireducibilna Perron-Frobenijusov teorem 1.2.14 bi garantirao jedinstvenost svojstvenog vektora, odnosno jedinstvenost vektora vrijednosti autoriteta.

Dakle, reducibilnost matrice A ili B je uzrok konvergencije prema nejedinstvenim rješenjima.

2.2 Primjer HITS metode

Prikažimo sada na primjeru HITS metodu. Za početak od korisnika dobivamo određene stavke upita. Naravno, u ovom primjeru nemamo ni približan red veličine matrice onom koji se koristi na webu. Recimo da stranice s oznakama 1 i 5 sadrže stavke korisnikovog upita. Zbog održavanja jednostavnosti primjera način na koji to radimo bit će opisan u sljedećem poglavlju 3.3. Uzmimo za primjer povezanog grafa prikaz strukture weba s početka poglavlja kojem smo promijenili oznake.

Prikaz weba, odnosno povezanog grafa kojeg ćemo koristiti možemo vidjeti na slici 2.3.



Slika 2.3: Graf susjedstva za vrhove 1 i 5

Za početak, trebamo odrediti matricu susjedstva $\mathbf{L}$ koja je definirana izrazom 2.2. Imamo:

$$\mathbf{L} = \begin{array}{c} \begin{matrix} 1 & 5 & 29 & 37 & 72 \end{matrix} \\ \begin{pmatrix} 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array} \begin{matrix} 1 \\ 5 \\ 29 \\ 37 \\ 72 \end{matrix} .$$

Dalje, izračunajmo matricu autoriteta A i hub matricu B.

$$A = \mathbf{L}^T\mathbf{L} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 1 \\ 1 & 1 & 2 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 2 \end{pmatrix}$$

$$B = \mathbf{L}\mathbf{L}^T = \begin{pmatrix} 2 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 2 & 1 & 0 \\ 1 & 1 & 1 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Za početni vektor uzmimo $\mathbf{x}^{(0)} = \mathbf{y}^{(0)} = (1/5 \quad 1/5 \quad 1/5 \quad 1/5 \quad 1/5)^T$. Sada koristeći

metodu potencija 1.1.6 s 1-normom imamo:

$$\mathbf{x}^{(\infty)} = (0.088247 \quad 0.283654 \quad 0.283654 \quad 0.088247 \quad 0.256198)^T$$
$$\mathbf{y}^{(\infty)} = (0.2039 \quad 0.1405 \quad 0.2039 \quad 0.4516 \quad 0)^T.$$

Prilikom računanja u praksi koristeći puno veće matrice nije izgledno dobiti iste vrijednosti u svojstvenom vektoru kao što smo mi dobili u vektoru hub vrijednosti. Redoslijed vrhova s istim vrijednostima autoriteta ili hub vrijednostima može se odabrati na proizvoljan način. U ovom slučaju mi ćemo dati prioritet stranici koja se prije nalazi u vektoru.

Konačno, koristeći dobivene vrijednosti sortiramo ih silazno te prikazujemo redoslijed pomoću oznaka weba.

$$\text{Poredak autoriteta} = (29 \quad 5 \quad 72 \quad 37 \quad 1)$$
$$\text{Poredak hubova} = (37 \quad 1 \quad 29 \quad 5 \quad 72)$$

Ovo označava kako stranica s oznakom 29 je najbolji autoritet za korisnikovog upit, a stranica s oznakom 37 najbolji hub za korisnikov upit.

2.3 Prednosti i mane HITS metode

Za početak jedna od očitih prednosti je dobivanje dva poretka web stranica. Ovisno o korisnikovim preferencijama on sam može birati hoće li veću važnosti pridodati poretku autoriteta ili poretku hubova. Nadalje, prednost HITS metode je što se proces pronalaženja tražene informacije na cijelom webu svodi na pronalaženje svojstvenog vektora relativno malih matrica. Naravno, ne malih kao u prethodnom primjeru nego relativno malih spram ogromne veličine weba.

Također, očiti nedostatak HITS metode je potreba za računanjem grafa susjedstva te pronalaska svojstvenih vektora dobivenih matrica za svaki upit.

Jedan od glavnih nedostataka je osjetljivost na *spamminga*. *Spamming*, u slučaju pretraživanja weba, označava manipulaciju pretraživača kako bi stranica bila bolje rangirana. Očita je pomisao ako vlasnik stranice zna da se prilikom pretraživanja koristi HITS metoda dodati brojne linkove na druge stranice i time očito utjecati na hub vrijednost svoje stranice. Zatim, jer se vrijednosti autoriteta i hub vrijednost računaju ovisno jedna o drugoj, porast hub vrijednosti rezultira i porastom vrijednosti autoriteta. Također, spomenuli smo kako je mala veličina grafa susjednosti prednost za brzinu računanja. Nažalost, zbog male veličine grafa susjedstva lokalne promjene linkova će uzrokovati velike promjene u poretku. Također, može doći i do skretanja s teme. Ukoliko stranice koje sadrže relevantne

stavke korisnikovog upita sadrže linkove za neku stranicu koja zapravo nema veze s korisnikovim upitom tada ta stranica poprima veliku vrijednost autoriteta. Zatim, zbog velike vrijednosti autoriteta može doći do poboljšanja poretka stranica koje također daju nebitne informacije za korisnikov upit.

Ipak, problem *spamming*-a HITS metode popravili su Henzinger i Bharat koristeći korak normalizacije, kojim se smanjuje relevantnost stranica koje imaju iznimno puno poveznica (što je zapravo glavna ideja PageRank metode što ćemo vidjeti u sljedećem poglavlju). Također, popravili su problem skretanja s teme HITS metode pridruživanjem određenih težina vrijednostima autoriteta i hub vrijednosti ovisno o stavkama upita [5].

2.4 Poveznica HITS-a i Bibliometricsa

Bibliometrics označava korištenje metoda za proučavanje knjiga, članaka i ostalih vrsta publikacija posebice znanstvenog sadržaja i obraća veliku pažnju na strukturu citiranja. Koristeći određene metode pomoću strukture citiranja knjiga i članaka moguće je napraviti određen poredak u važnosti i utjecaju publikacija. U radovima [7, 8] Ding i suradnici opisuju vezu između HITS metode te dva osnovna koncepta u bibliometricsu, *co-citation* i *co-reference*.

Co-reference označava pojavu kada dva različita dokumenta citiraju neki treći dokument. *Co-citation* označava pojavu kada su dva različita dokumenta citirana od istog trećeg dokumenta. U navedenim radovima pokazano je da matrica autoriteta je izravno povezana s *co-citation*-om, dok je hub matrica izravno povezana s *co-reference*-om.

Za primjer uzmimo prethodno korištenu matricu susjedstva grafa 2.3.

$$\mathbf{L} = \begin{pmatrix} 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

Prethodno smo izračunali i matricu autoriteta i hub matricu

$$A = \mathbf{L}^T\mathbf{L} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 1 \\ 1 & 1 & 2 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 2 \end{pmatrix}, \quad B = \mathbf{L}\mathbf{L}^T = \begin{pmatrix} 2 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 2 & 1 & 0 \\ 1 & 1 & 1 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

Ding i suradnici [7, 8] pokazuju da matricu autoriteta možemo zapisati kao $A = D_{in} + C$, gdje je D_{in} dijagonalna matrica koja na dijagonali sadrži broj linkova na stranicu za pojedini

vrh (broj ”dolaznih” linkova). Primjerice, matrica D_{in} na poziciji $(2, 2)$ ima broj 2 što znači da postoje dvije stranice koje imaju link na stranicu s oznakom 5. C_{ij} označava broj stranica koje imaju link na stranicu koja odgovara i-tom retku i na stranicu koja odgovara j-tom stupcu. Na primjer, $C_{32} = 1$ označava da postoji jedna stranica koja ima na link na stranice s oznakama 5 i 29. Na slici 2.3 možemo vidjeti da je to stranica s oznakom 37.

Zatim, pokazano je i da hub matricu možemo zapisati kao $B = D_{out} + R$ gdje je D_{out} dijagonalna matrica koja na dijagonali sadrži broj linkova sa stranice za pojedini vrh (broj ”odlaznih” linkova). Primjerice, matrica D_{out} na poziciji $(4, 4)$ ima broj 3 što znači da postoje tri linka sa stranice koja ima oznaku 37. R_{ij} označava broj stranica na koje imaju poveznice i stranica koja odgovara i-tom retku i stranica koja odgovara j-tom stupcu. Na primjer, $R_{34} = 1$ označava da postoji jedna stranica na koju imaju poveznice stranice s oznakama 29 i 37. Na slici 2.3 možemo vidjeti da je to stranica s oznakom 5.

Poglavlje 3

PageRank

Sljedeća metoda za pretraživanje weba koju obrađujemo je PageRank. PageRank je nastao iste 1998. godine kao i HITS. PageRank metodu su razvili osnivači Google-a Larry Page i Sergey Brin koju su koristili za osnovu njihovog pretraživača. Dakle, PageRank je bio prvi algoritam koji je Google koristio i još uvijek je najpoznatiji Google-ov korišteni algoritam za pretraživanje koji je najvjerojatnije i zaslužan za postanak Google-a kao svjetskog najpoznatijeg pretraživača. Od rujna 2019. PageRank i njemu povezani patenti su istekli.

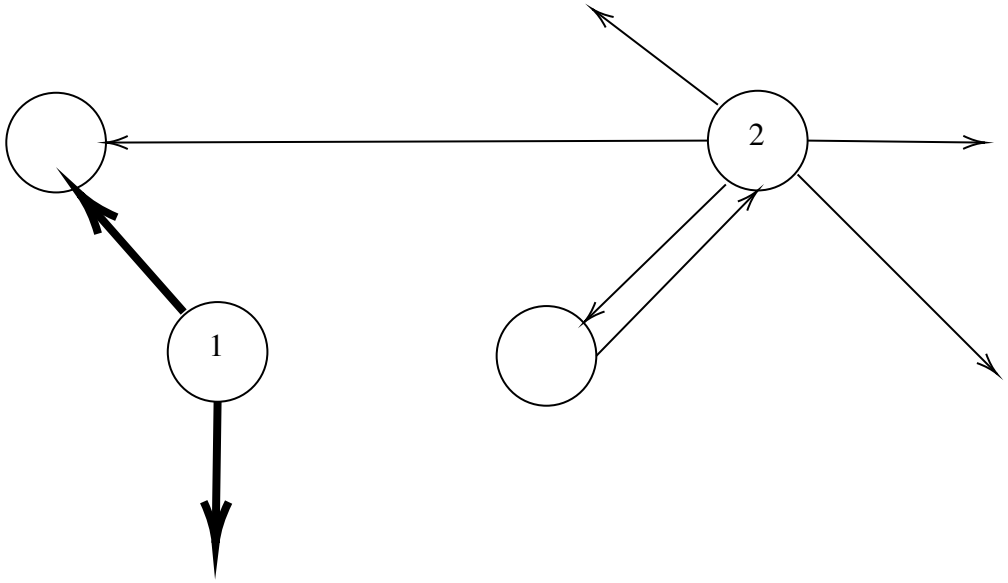
Za razliku od HITS metode, gdje smo spomenuli kako je jedan od većih nedostatak neprestano računanje svojstvenih vektora te time i poretka, u PageRank metodi svaka stranica ima izračunatu takozvanu PageRank vrijednost kojom određujemo prioritet stranica. Također, za razliku od HITS metode ideja PageRank metode je da svi linkovi ne nose istu težinu. Primjerice, link s bitne stranice nosi veću težinu nego link s manje bitne stranice. Također, važnost stranice bi trebala biti propisno skalirana ovisno o broju stranica na koje ima linkove. Na slici 3.1 vidimo kako stranica s oznakom 2 ima puno više poveznica nego stranica s oznakom 1. Ideja PageRank-a je, kao što je nacrtano, pridodati veće značenje poveznicama sa stranice s oznakom 1 nego stranice sa oznakom 2.

Ovim razmišljanjem je nastala sljedeća definicija PageRank vrijednosti $r(P)$ stranice P:

$$r(P) = \sum_{Q \in B_P} \frac{r(Q)}{\left|Q\right|},$$

gdje B_P označava sve stranice koje imaju poveznicu prema P te $\left|Q\right|$ broj linkova sa stranice Q.

Ovo je očito rekurzivna definicija. Za izračunavanje PageRank vrijednosti potrebna nam je iterativna verzija. Iterativnu verziju dobivamo tako da početnu PageRank vrijednost svih stranica definiramo kao $r_0(P_i) = 1/n$, gdje i označava oznaku stranice, a n ukupan broj stranica. Očito ovo definiramo za svaki i između 1 i n. Sada dobivamo iterativnu definiciju



Slika 3.1: PageRank primjer

PageRank vrijednosti izrazom:

$$r_j(P_i) = \sum_{Q \in B_P} \frac{r_{j-1}(Q)}{|Q|}, \quad j = 1, 2, \ldots$$

Koristeći vektorski zapis, gdje označavamo $\pi_j^T = (r_j(P_1), \ldots, r_j(P_n))$, imamo sljedeći izraz:

$$\pi_j^T = \pi_{j-1}^T \mathbf{P},$$

gdje je $\mathbf{P}$ matrica definirana na sljedeći način:

$$p_{ij} = \begin{cases} 1/|P_i|, & \text{ako postoji link sa stranice } i \text{ na stranicu } j \\ 0, & \text{inače} \end{cases}.$$

Možemo već ovdje primijetiti kako PageRank također koristi metodu potencija.
Konačno, ako limes postoji, PageRank vektor definira se kao:

$$\pi^T = \lim_{j \to \infty} \pi_j^T$$

Napomenimo da i-ta komponenta vektora označava PageRank vrijednost stranice P_i.

U sljedećim odjeljcima govorimo o računanju i ažuriranju PageRank vektora, konvergenciji metode te prednostima i manama PageRank metoda posebice u usporedbi s HITS metodom.

3.1 Markovljev model weba

Iako smo htjeli na jednom mjestu imati svu teoriju složenu u prvom poglavlju ovaj dio odlučujemo ovdje napisati jer je usko vezan isključivo s PageRank metodom. Primijetimo da prethodno definirana matrica **P** ima retke sume nula ili jedan. Retci sume nule označavaju one stranice koje nemaju poveznice na druge stranice weba. Pretpostavljamo da ovdje takvih stranica nema. Tada je matrica **P** stohastička.

Definicija 3.1.1. *Matrica* $\mathbf{P} = \left(p_{ij}\right)$ *je stohastička matrica ako je* $p_{ij} \geq 0$ *za sve* i, j *te*

$$\sum_{j=1}^{n} p_{ij} = 1, \quad \text{za sve } 1 \leq i \leq n.$$

Nadalje, PageRank se zapravo bazira na *random* surferu. *Random* surfer označava korisnika koji nasumično klika na poveznice s jedne stranice na drugu. Taj proces možemo interpretirati koristeći Markovljeve lance. Definicije vezane za Markovljeve lance smo preuzeli iz skripte [22].

Definicija 3.1.2. *Neka je* S *skup. Slučajan proces s diskretnim vremenom i prostorom stanja* S *je familija* $X = (X_n : n \geq 0)$ *slučajnih varijabli (ili elemenata) definiranih na nekom vjerojatnosnom prostoru* $(\Omega, \mathcal{F}, \mathbb{P})$ *s vrijednostima u* S*. Dakle, za svaki* $n \geq 0$*, je* $X_n : \Omega \to S$ *slučajna varijabla.*

Definicija 3.1.3. *Neka je* S *prebrojiv skup. Slučajni proces* $X = (X_n : n \geq 0)$ *definiran na vjerojatnosnom prostoru* $(\Omega, \mathcal{F}, \mathbb{P})$ *s vrijednostima u skupu* S *je Markovljev lanac ako vrijedi*

$$\mathbb{P}(X_{n+1} = j \mid X_n = i, X_{n-1} = i_{n-1}, ..., X_0 = i_0) = \mathbb{P}(X_{n+1} = j \mid X_n = i) \tag{3.1}$$

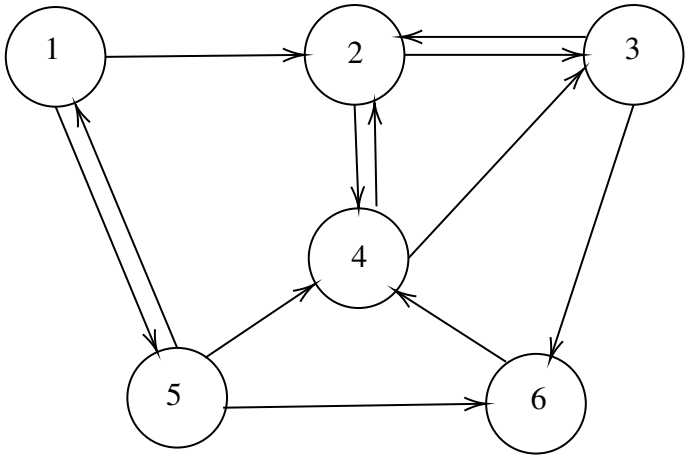
za svaki $n \geq 0$ *i za sve* $i_0, ...i_{n-1}, i, j \in S$ *za koje su obje uvjetne vjerojatnosti dobro definirane.*

Ovdje n predstavlja vremenski trenutak. Stoga, vrijeme $n + 1$ predstavlja neposrednu budućnost, a vremena $0, 1, \ldots n - 1$ predstavljaju prošlost.

U slučaju PageRank-a ne želimo da nam desna strana jednadžbe 3.1 ovisi o vremenu $n \geq 1$. Stoga, uvodi se pojam homogenog Markovljevog lanca.

Definicija 3.1.4. *Neka je* $\lambda = (\lambda_i : i \in S)$ *vjerojatnosna distribucija na* S*, te neka je* $P = (p_{ij} : i, j \in S)$ *stohastička matrica.. Slučajni proces* $X = (X_n : n \geq 0)$ *definiran na vjerojatnosnom prostoru* $(\Omega, \mathcal{F}, \mathbb{P})$ *s vrijednostima u skupu* S *je homogen Markovljev lanac s početnom distribucijom* λ *i prijelaznom matricom* P *ako vrijedi*

$$P(X_0 = i) = \lambda_i \quad \text{za sve } i \in S$$
$$\mathbb{P}(X_{n+1} = j \mid X_n = i, X_{n-1} = i_{n-1}, ..., X_0 = i_0) = p_{ij}$$



Slika 3.2: Primjer povezanog grafa sa 6 čvorova

za svaki $n \geq 0$ *i za sve* $i_0, ...i_{n-1}, i, j \in S$.

Pokažimo kako Markovljev model predstavlja graf povezanosti na primjeru. Promotrimo sliku 3.2.

Markovljev model predstavlja graf povezanosti weba kao prijelaznu vjerojatnosnu matricu $\mathbb{P}$ gdje p_{ij} označava vjerojatnost prijelaza iz stanja i u stanje j, odnosno vjerojatnost prijelaza iz vrha i u vrh j, u jednom koraku. Ako pretpostavimo da je jednako vjerojatno iz jednog vrha preći u drugi (jednako vjerojatno kliknuti na bilo koju poveznicu s trenutne stranice) tada dobivamo prethodnu definiranu matricu

$$\mathbf{P} = \begin{pmatrix} 0 & 0.5 & 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0.5 & 0.5 & 0 & 0 \\ 0 & 0.5 & 0 & 0 & 0 & 0.5 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0.33 & 0 & 0 & 0.33 & 0 & 0.33 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}.$$

Ipak, ako se primijeti da korisnici sa stranice s oznakom 5 idu tri puta češće na stranicu s oznakom 1 nego na stranicu s oznakom 4 i stranicu s oznakom 6, tada možemo alternativno definirati peti red kao

$$\mathbf{p}_5^T = (0.75 \quad 0 \quad 0 \quad 0.25 \quad 0 \quad 0.25).$$

U radu [20] je predloženo na koji način se može alternirati matrica kako bi PageRank poredak za specifične teme bio ispravniji. Također, tamo opisanom formulom matrica $\mathbf{P}$ je ireducibilna što je iznimno poželjno zbog jedinstvenosti PageRank vektora. Ireducibilnosti

ćemo se konkretnije dotaknuti malo kasnije. U radu [24] se smatra kako bi algoritam trebao uključivati sljedeće metrike: relevantnost, autoritet, integrativnost i novitet. Opisan je određeni način modificiranja matrice kako bi zadovoljili navedene metrike te koji bi osigurao ireducibilnost matrice **P**.

Nadalje, vrijedi sljedeća propozicija:

Propozicija 3.1.5. *Spektralni radijus stohastičke matrice je* 1.

Dokaz. Prvo, primijetimo da je 1 svojstvena vrijednost proizvoljne stohastičke matrice **P** sa svojstvenim vektorom **e**, gdje **e** označava vektor jedinica. Dakle, jer je sprektralni radijus po definiciji najveća svojstvena vrijednost po apsolutnoj vrijednosti slijedi $1 \leq \rho(\mathbf{P})$.

U teoremu 1.2.2 smo pokazali kako je spektralni radijus matrice manji ili jednak proizvoljnoj matričnoj normi iste te matrice. Posebno, za normu beskonačno, za koju podsjećamo da iznosi maksimalnoj sumi elemenata po redu, vrijedi:

$$\begin{aligned}\rho(\mathbf{P}) &\leq \|\mathbf{P}\|_\infty \\ &= \max_{1\leq 1\leq n} \sum_{j=1}^{n} p_{ij} \\ &= \max_{1\leq 1\leq n} 1 \\ &= 1.\end{aligned}$$

Dakle, vrijedi $\rho(A) = 1$. □

Iz prethodne propozicije slijedi da ako PageRank iteracije konvergiraju, konvergiraju prema vektoru π^T koji zadovoljava sljedeće jednakosti:

$$\pi^T = \pi^T \mathbf{P}, \quad \pi^T \mathbf{e} = 1, \tag{3.2}$$

što je stacionarna distribucija Markovljevog lanca. Stacionarna distribucija Markovljevog lanca je vjerojatnosna distribucija koja ostaje ista u Markovljevom lancu za bilo koje odvijanje vremena. Ovo sve opravdava Google-ovu karakterizaciju PageRank iznosa kao udio vremena potrošenog na pojedinoj stranici web surfera koji beskonačno dugo i nasumično klika poveznice na stranici. Dodatno o Markovljevim lancima i stohastičkim matricama gdje se promatraju općenita svojstva istih, a ne specifična svojstva za PageRank koja mi promatramo, može se pronaći u posljednjem poglavlju knjige Meyera [17].

Konačno, prilazeći problemu na ovaj način preostaje rješiti jednadžbe 3.2, što je zapravo pronalazak svojstvenog vektora. To ne bi bio problem da se ne radi o 130 bilijuna stranica (ovaj broj je objavljen 2016. godine na njihovoj *How Search Works* stranici [1]) koja svaka ima svoj red u matrici. Za vrijeme pisanja u slučaju traženja najveće operacije s matricama svi rezultati će upravo ukazivati na ovaj problem. Stoga i ne čude navodi da je vrijeme potrebno za izračunavanje PageRank vektora trajalo i po nekoliko dana.

3.2 Modificiranje PageRank matrice

Već smo ranije spominjali moguće probleme za konvergenciju metode. Jedan od očitijih problema koji smo ne baš toliko neprimjetno preskočili je problem ukoliko stranica nema poveznice na druge. U prošlom koraku smo sva razmišljanja temeljili na pretpostavci da takve ne postoje. No očito to nije slučaj u stvarnosti. Za takve stranice odgovarajući red matrice će imati samo nule. Stoga, matrica $\mathbf{P}$ neće biti stohastička. Ipak ovaj problem možemo relativno lako riješiti. Svaki nul red mijenjamo s redom koji sadrži elemente $1/n$ gdje n označava broj stranica weba. Primjerice, imamo:

$$\mathbf{P} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/2 & 1/2 \\ 0 & 0 & 1/2 & 0 & 1/2 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix} \rightarrow \tilde{\mathbf{P}} = \begin{pmatrix} 1/5 & 1/5 & 1/5 & 1/5 & 1/5 \\ 1/5 & 1/5 & 1/5 & 1/5 & 1/5 \\ 0 & 0 & 0 & 1/2 & 1/2 \\ 0 & 0 & 1/2 & 0 & 1/2 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Ovakva modifikacija je kao da smo dodali stranicama bez poveznica poveznice na sve stranice weba. Ovime smo dobili stohastičku matricu $\tilde{\mathbf{P}}$. No, kao što vidimo matrica $\tilde{\mathbf{P}}$ je po definiciji reducibilna.

Reducibiline matrice nastaju ukoliko su proizašle iz reducibilinog Markovljevog lanca.

Definicija 3.2.1. *Kažemo da je Markovljev lanac reducibilan ako postoji pravi podskup stanja u kojem lanac ostane zauvijek, tj. nije moguće da lanac postigne neko drugo stanje izvan tog podskupa.*
Kažemo da je Markovljev lanac ireducibilan ako nije reducibilan.

Kao što smo već nekoliko puta spominjali ireducibilinost je poželjno svojstvo kako bi Perron-Frobenijusov teorem 1.2.14 garantirao jedinstvenost rješenja, u ovom slučaju je to vektor π^T.

Dakle, sada želimo matricu učiniti ireducibilnom. Očiti način za izbjegavanja reducibilnosti je izbacivanje svih nula. Stoga, jedan od pristupa pravljenja matrice $\tilde{\mathbf{P}}$ ireducibilnom, koji je bio i korišten u početku, je korištenjem matrice :

$$\mathbf{E} = \begin{pmatrix} 1/n & \dots & 1/n \\ \vdots & \ddots & \vdots \\ 1/n & \dots & 1/n \end{pmatrix}.$$

Zatim, jer je poželjno da matrica ostane stohastička potrebno je uvesti određeni skalar koji će to osigurati. Konačno, dobivamo matricu $\tilde{\tilde{\mathbf{P}}}$ koja je stohastička i ireducibilna.

$$\tilde{\tilde{\mathbf{P}}} = \alpha\tilde{\mathbf{P}} + (1-\alpha)\mathbf{E}$$

Ovo se može postići tako da se omogući direktan prijelaz iz proizvoljna dva stanja. U web pretraživanju ovakav model se može opravdati mogućnošću prelaska korisnika s jedne na bilo koju drugu stranicu koristeći URL adresu. Ako se radi o *random* surferu, tada je očito jednaka vjerojatnost unosa bilo kojeg URL-a i ima smisla koristiti matricu $\mathbf{E}$.

Kasnije, nakon ovog pristupa Google se odlučio za realniji pristup. U ovom pristupu odlučili su napraviti svoju neuniformnu distribuciju za odabir poveznice. Na ovaj način mogu dati veću važnost kvalitetnijim stranicama, a uostalom na ovaj način mogu iz komercijalnih razloga poboljšati, odnosno smanjiti PageRank iznos pojedinih stranica. To su postigli definiranjem matrice $\mathbf{E}$ kao

$$\mathbf{E} = \mathbf{e}\mathbf{v}^T,$$

gdje je $\mathbf{v}^T > 0$ takozvani vektor personilizacije.

Matrica $\tilde{\tilde{\mathbf{P}}}$ koristeći prethodno definiranu matricu $\mathbf{E} = \mathbf{e}\mathbf{v}^T$ naziva se Google-vom matricom. Jer je matrica ireducibilna i stohastička za $\tilde{\tilde{\mathbf{P}}}$ možemo pronaći jedinstveni vektor π^T, koji zapravo označava stacionarnu distribuciju. Vektor π^T nazivamo PageRank vektorom. Očito, moguće su i ostale metode forsiranja ireducibilnosti, pogotovo bez mijenjenja svih elemenata vrijednosti nula u koje nećemo dodatno zalaziti. Iako se čini pretjerano na ovaj umjetan način povezati sve stranice, Google je ostao pri ovom načinu.

Prilikom dokazivanja metode potencija 1.1.6 mogli smo primijetiti kako brzina konvergencije ovisi znatno o omjeru spektralnog radijusa i sljedeće najveće svojstvene vrijednosti po apsolutnoj vrijednosti.

Možemo pokazati da za spektre $\sigma\left(\tilde{\mathbf{P}}\right) = \{1, \mu_2, \ldots \mu_n\}$ i $\sigma\left(\tilde{\tilde{\mathbf{P}}}\right) = \{1, \lambda_2, \ldots \lambda_n\}$ vrijedi:

$$\lambda_k = \alpha\mu_k \quad \text{za } k = 2, 3, \ldots, n$$

neovisno o vektoru $\mathbf{v}^T$.

Dokaz. [17] Prvo, pokažimo da vrijedi

$$\det\left(\mathbf{I} + \mathbf{c}\mathbf{d}^T\right) = 1 + \mathbf{d}^T\mathbf{c}. \tag{3.3}$$

Tvrdnja slijedi iz sljedeće jednakosti:

$$\begin{pmatrix} \mathbf{I} & 0 \\ \mathbf{d}^T & 1 \end{pmatrix}\begin{pmatrix} \mathbf{I} + \mathbf{c}\mathbf{d}^T & \mathbf{c} \\ 0 & 1 \end{pmatrix}\begin{pmatrix} \mathbf{I} & 0 \\ -\mathbf{d}^T & 1 \end{pmatrix} = \begin{pmatrix} \mathbf{I} & \mathbf{c} \\ 0 & 1 + \mathbf{d}^T\mathbf{c} \end{pmatrix}.$$

Korištenjem svojstva umnoška determinanti slijedi:

$$\det\begin{pmatrix} \mathbf{I} & 0 \\ \mathbf{d}^T & 1 \end{pmatrix}\det\begin{pmatrix} \mathbf{I} + \mathbf{c}\mathbf{d}^T & \mathbf{c} \\ 0 & 1 \end{pmatrix}\det\begin{pmatrix} \mathbf{I} & 0 \\ -\mathbf{d}^T & 1 \end{pmatrix} = \det\begin{pmatrix} \mathbf{I} & \mathbf{c} \\ 0 & 1 + \mathbf{d}^T\mathbf{c} \end{pmatrix}$$

$$1 \cdot \det\left(\mathbf{I} + \mathbf{c}\mathbf{d}^T\right) \cdot 1 = 1 + \mathbf{d}^T\mathbf{c}.$$

Pokažimo sada da vrijedi za $\lambda \notin \sigma\left(\tilde{\mathbf{P}}\right)$:

$$\left(\tilde{\mathbf{P}} - \lambda\mathbf{I}\right)^{-1}\mathbf{e} = \frac{\mathbf{e}}{1-\lambda}. \tag{3.4}$$

To možemo vidjeti na sljedeći način:

$$\mathbf{e} = \left(\tilde{\mathbf{P}} - \lambda\mathbf{I}\right)^{-1}\left(\tilde{\mathbf{P}} - \lambda\mathbf{I}\right)\mathbf{e} = \left(\tilde{\mathbf{P}} - \lambda\mathbf{I}\right)^{-1}\left(\tilde{\mathbf{P}}\mathbf{e} - \lambda\mathbf{e}\right) = \left(\tilde{\mathbf{P}} - \lambda\mathbf{I}\right)^{-1}(1-\lambda)\,\mathbf{e}.$$

Također, trebamo pokazati i da vrijedi:

$$\det\left(\tilde{\mathbf{P}} + \mathbf{e}\mathbf{v}^T\right) = \det\left(\tilde{\mathbf{P}}\right)\left(1 + \mathbf{v}^T\tilde{\mathbf{P}}^{-1}\mathbf{e}\right). \tag{3.5}$$

Zapišimo $\tilde{\mathbf{P}} + \mathbf{e}\mathbf{v}^T$ kao $\tilde{\mathbf{P}}\left(\mathbf{I} + \tilde{\mathbf{P}}^{-1}\mathbf{e}\mathbf{v}^T\right)$. Koristeći 3.3 imamo:

$$\begin{aligned}\det\left(\tilde{\mathbf{P}}\left(\mathbf{I} + \tilde{\mathbf{P}}^{-1}\mathbf{e}\mathbf{v}^T\right)\right) &= \det\left(\tilde{\mathbf{P}}\right)\det\left(\mathbf{I} + \tilde{\mathbf{P}}^{-1}\mathbf{e}\mathbf{v}^T\right)\\ &= \det\left(\tilde{\mathbf{P}}\right)\left(1 + \mathbf{v}^T\tilde{\mathbf{P}}^{-1}\mathbf{e}\right).\end{aligned}$$

Konačno, pokažimo traženu tvrdnju. Krenimo od zapisa karakterističnog polinoma matrice $\tilde{\tilde{\mathbf{P}}}$. Imamo:

$$\begin{aligned}\det\left(\tilde{\tilde{\mathbf{P}}} - \lambda\mathbf{I}\right) &= \det\left(\alpha\tilde{\mathbf{P}} + (1-\alpha)\mathbf{e}\mathbf{v}^T - \lambda\mathbf{I}\right)\\ &\overset{3.5}{=} \det\left(\alpha\tilde{\mathbf{P}} - \lambda\mathbf{I}\right)\left(1 + (1-\alpha)\mathbf{v}^T\left(\alpha\tilde{\mathbf{P}} - \lambda\mathbf{I}\right)^{-1}\mathbf{e}\right)\\ &\overset{3.4}{=} \left(\pm\prod_{k=1}^{n}(\alpha\mu_k - \lambda)\right)\left(1 + (1-\alpha)\frac{\mathbf{v}^T\mathbf{e}}{\alpha - \lambda}\right)\\ &= \left(\pm\prod_{k=2}^{n}(\alpha\mu_k - \lambda)\right)\left(\alpha + (1-\alpha)\mathbf{v}^T\mathbf{e} - \lambda\right).\end{aligned}$$

Kao što vidimo za $k \neq 1$ slijedi $\lambda_k = \alpha\mu_k$. Iz druge zagrade pak imamo:

$$\begin{aligned}\lambda &= \alpha + (1-\alpha)\underbrace{\mathbf{v}^T\mathbf{e}}_{1}\\ &= \alpha + 1 - \alpha\\ &= 1.\end{aligned}$$

Ovaj λ zapravo označava preostalu svojstvenu vrijednost $\lambda_1 = 1$. □

Kao što smo prethodno komentirali, brzina konvergencije ovisi o omjeru dvije, po apsolutnu najvećoj vrijednosti, svojstvene vrijednosti. Zbog strukture matrice $\tilde{\mathbf{P}}$ izgledno je da $\mu_2 = 1$ ili je približan 1. Stoga, po prethodnom dokazu možemo s koeficijentom α utjecati na vrijednosti λ_2. Na prvu se čini pametno uzeti što manji α kako bi algoritam brže konvergirao. Ipak, ne smijemo zaboraviti da α označava na neki način koliko ćemo modificirati matricu $\tilde{\mathbf{P}}$ matricom $\mathbf{E}$. Dakle, potrebno je pronaći ravnotežu između brze konvergencije i što manje modifikacije matrice k ireducibilnoj. Google je originalno izjavio kako koriste $\alpha = 0.85$ što izgledno daje $\lambda = 0.85$.

Zatim, lako se vidi da 114 iteracija algoritma 1.1.6 daje preciznost od $(0.85)^{114} < 10^{-8}$ za iznos PageRank vrijednosti stranica, što im je u to vrijeme izgleda bilo i više nego dovoljno.

3.3 PageRank implementacija

U prošlom poglavlju prilikom opisivanja HITS metode prešutno smo prešli preko prvog dijela metode. Mislimo na dio kada za određene stavke upita dobivamo stranice koji sadrže te stavke. Taj dio metode je identičan kao i za PageRank metode i čini se kako ima više smisla ga objasniti sada nego ranije. To se može napraviti metodom koja koristi invertiranu stavka-dokument datoteku. U toj datoteci svakoj mogućoj stavci pridružujemo sve dokumente, odnosno sve oznake dokumenata koji je sadrže. Primjerice, datoteka bi mogla izgledati na sljedeći način:

- abeceda : stavka 1 - dokumenti $4, 29, 3092, 137$

$\vdots$

- oaza : stavka 315 - dokumenti $2, 251, 29, 1102$

$\vdots$

- žutilo : stavka m - dokumenti $1322, 14, 901$

Sada na primjer za korisnikov upit koji sadrži stvaku 315 i stavku m za dokumente s oznakama 2, 251, 29, 1102, 1322, 14, 1901 pravimo prethodno opisani graf susjedstva za HITS metodu te prelazimo na drugi dio metode što je metoda potencija, a za PageRank metodu jednostavno sortiramo dobivene dokumente po iznosu unaprijed izračunate PageRank vrijednosti. Naravno, Google nije koristio isključivo PageRank vrijednost za određivanje prioriteta između dokumenata.

Posebno naglasimo da se ovisno o korisnikovom upitu za HITS metodu računaju vrijednosti autoriteta i hub vrijednosti, dok za PageRank metodu korisnikov upit **ne** utječe na PageRank vrijednost.

Dakle, cijela PageRank metoda se svodi na računanje svojstvenog vektora π^T matrice veličine reda broja svih stranica na webu pomoću metode potencija, gdje ireducibilnost matrice zbog Perron-Frobenijusovog teorema osigurava jedinstvenost tog rješenja. Ovaakav način implementacije izvodi se brže zbog nužnog paralelnog izvođenja metode kojeg nećemo posebno promatrati. Brin i Page, osnivači Google-a, koristeći relativno jednostavan algoritam metode potencija su izjavili da već za 50 iteracija matrice reda $n = 322000000$ su dobili korisne rezultate.

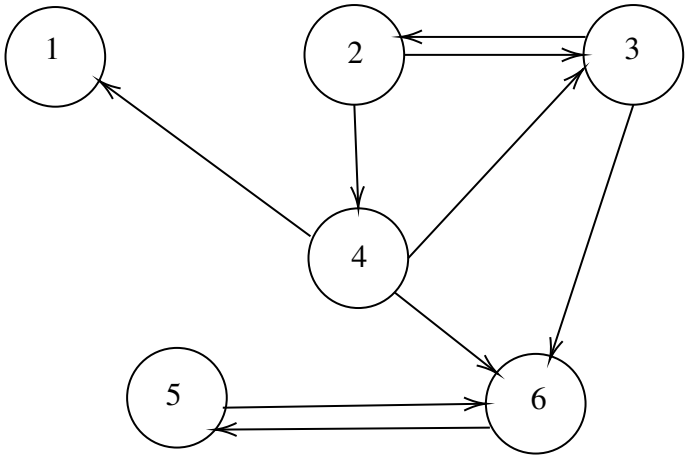
Naravno, nakon osnovne PageRank metode nije dugo trebalo da znanstvenici pronađu poboljšanja iste. Većina tih istraživanja temelji se na poboljšanje metode potencija koja uzrokuju bržu konvergenciju metode. Neka od njih mogu se pronaći u sljedećim radovima [4, 10, 12, 25].

Problem ovakve implementacije je preciznost PageRank vrijednosti. Ne zna se točno koliku razinu točnosti zahtijeva Google, ali mora biti dovoljno velika kako bi mogla razlikovati PageRank vrijednosti liste stranica koje su relevantne za upit. Podsjećamo kako je π^T zapravo vektor distribucije. Dakle pojedini π_i se nalazi između 0 i 1. Za vrijeme pisanja rada [14] postojalo je približno 7 milijardi stranica. Stoga, najmanje red veličine 10^{-9} je dovoljan za međusobnu usporedbu pojedinih vrijednosti. Iako se pojedine PageRank vrijednosti stranica mogu razlikovati tek na "nižim" decimalama nije izgledno da će se u skupu stranica vraćenih za određeni upit dogoditi takav slučaj. Već ranije spomenuto, po zadnjim dostupnim podatcima iz 2016. godine broj stranica je 130 bilijuna. Dakle, preciznost reda na 10^{-9} gotovo sigurno nije bila dostatna.

Određena promatranja sugeriraju da prirodna struktura weba čini NCD (*nearly completely decomposable*) Markovljev lanac. NCD Markovljev lanac je Markovljev lanac gdje se skup stanja može particionirati na takav način da su prijelazi unutar particije puno češći nego prijelazi između različitih particija. Za pretraživanje weba to znači da kada nasumično odabiremo poveznice izgledno je da ćemo "ostajati" u određenoj grupi (particiji) stranica te da su prijelazi u drugu grupu (particiju) stranica vrlo malo izgledni. Više o NCD Markovljevim lancima možete pronaći u [13]. Povećanjem značaja matrice $\mathbf{E}$ kojom se osigurava ireducbilnost te time jedinstvenost može doći do prikrivanja NCD svojstva Markovljevog lanca. Značajan rad vezan za računanje stacionarnog vektora NCD sustava je napravio Williama J. Stewarta u knjizi [21] te brojnim ostalim publikacijama. Stoga, pokazivanjem NCD svojstva Markovljevog lanca može doprinijeti novim pristupima implementacije PageRank metode poput *NCDawareRank*-a [19].

Promotrimo sada ažuziranje PageRank vrijednosti. Kao što smo već rekli računanje vektora π^T može trajati nekoliko dana pa nema smisla ažužirati cijelu matricu zbog dodavanja neke poveznice. Stoga, Google je izjavio kako je ažuriranje izvodio svaka dva tjedna. Nažalost, način računanja vektora π^T nam ne može biti od pretjerane pomoći prilikom ponovnog računanja. Dakle, Google nakon svakog ažuriranja kreće isponova.

Postoje istraživanja o ažuriranju PageRank vektora koristeći posljednje izračunati



Slika 3.3: Primjer weba sa 6 čvorova

PageRank vektor i promjene u strukturi weba kojima ne bi bilo potrebno izračunavanje iz nule [15]. Ipak, ovo je moguće jedino prilikom dodavanja, brisanja i mijenjanja poveznica. Prilikom dodavanja ili brisanja stranica mijenja se dimenzija matrice i očito je to znatno veći problem nego s poveznicama.

Također, postoji nekoliko algoritama za ažuriranje Markovljevih lanaca [15], ali oni ne rade nažalost dobro za PageRank metodu. Ipak, napravljeni su algoritmi koji su prilagođeni dodavanju/brisanju i poveznica i stranica. Mogu se izdvojiti iterativna agregacijska tehnika [15] i adaptivna akceleracija metode potencija [11]. Također, aktivna područja u kojima se istražuje poboljšanje su *clustering* i paralelizacija.

3.4 Primjer PageRank metode

Napravimo sada primjer PageRank metode. Primjer ćemo napraviti na web-u sa 6 čvorova sličan onom s početka poglavlja. Prikaz weba možemo vidjeti na slici 3.4.

Zatim određujemo takozvanu Google-ovu matricu **P**.

$$\mathbf{P} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/2 & 1/2 & 0 & 0 \\ 0 & 1/2 & 0 & 0 & 0 & 1/2 \\ 1/3 & 0 & 1/3 & 0 & 0 & 1/3 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Sljedeći korak je dobiti stohastičku matricu $\tilde{\mathbf{P}}$. Matrica **P** nije stohastička jer stranica s oznakom 1 nema poveznica pa je suma prvog reda 0, a ne 1. Ovo lagano popravljamo

dodavanjem $1/6$ umjesto 0 u prvom redu. Ovo je kao da smo sa stranici s oznakom 1 napravili poveznice prema ostalim stranicama. Imamo:

$$\tilde{\mathbf{P}} = \begin{pmatrix} 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 \\ 0 & 0 & 1/2 & 1/2 & 0 & 0 \\ 0 & 1/2 & 0 & 0 & 0 & 1/2 \\ 1/3 & 0 & 1/3 & 0 & 0 & 1/3 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}.$$

Matrica je sada stohastička, ali je reducibilna po definiciji 1.2.9. Kako bi iskoristili Perron-Frobenijusov teorem 1.2.14 koji bi osigurao jedinstvenost stacionarne distribucije/PageRank vektora π^T moramo imati ireducibilnu matricu. Ireducibilnost dobivamo na prethodno opisani način dodavanjem skalirane matrice $\mathbf{E}$. Uzmimo da je $\alpha = 0.9$. Tada vrijedi $\tilde{\tilde{\mathbf{P}}} = 0.9\tilde{\mathbf{P}} + (1 - 0.9)\mathbf{E}$.
Imamo:

$$\tilde{\tilde{\mathbf{P}}} = 0.9 \cdot \begin{pmatrix} 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 \\ 0 & 0 & 1/2 & 1/2 & 0 & 0 \\ 0 & 1/2 & 0 & 0 & 0 & 1/2 \\ 1/3 & 0 & 1/3 & 0 & 0 & 1/3 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} + 0.1 \cdot \begin{pmatrix} 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 \\ 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 \\ 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 \\ 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 \\ 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 \\ 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 \end{pmatrix}$$

$$= \begin{pmatrix} 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 \\ 1/60 & 1/60 & 7/15 & 7/15 & 1/60 & 1/60 \\ 1/60 & 7/15 & 1/60 & 1/60 & 1/60 & 7/15 \\ 19/60 & 1/60 & 19/60 & 1/60 & 1/60 & 19/60 \\ 1/60 & 1/60 & 1/60 & 1/60 & 1/60 & 11/12 \\ 1/60 & 1/60 & 1/60 & 1/60 & 11/12 & 1/60 \end{pmatrix}.$$

Matrica je sada ireducibilna. Dakle, po Perron-Frobenijusovom teoremu PageRank vektor π^T je jedinstven.

Koristeći standardnu metodu potencija dobivamo:

$$\pi^T = (0.034812 \quad 0.047089 \quad 0.056002 \quad 0.043078 \quad 0.399475 \quad 0.419541).$$

Napominjemo da ovaj račun do sada nema nikakve veze s korisnikovim upitom. Pretpostavimo da korisnikov upit sadrži stavke 12 i 8. Nadalje, pretpostavimo da se stavka 12 nalazi na stranicama s oznakama 1, 3 i 5, a stavka 8 samo na stranici s oznakom 4. Ovi podaci se nalaze u invertiranoj stavka-dokument datoteci čiji izgled smo opisali na početku

odjeljka 3.3. Preostaje sortirati vrijednosti PageRank vrijednosti odabranih stranica.

$$\pi_5 = 0.364$$
$$\pi_3 = 0.076$$
$$\pi_4 = 0.059$$
$$\pi_1 = 0.048.$$

Dakle, redoslijed prikaza stranica na korisnikov upit je stranica s oznakom 5, stranica s oznakom 3, stranica s oznakom 4 te konačno stranica s oznakom 1.

Opet napominjemo da u slučaju novog korisnikovog upita potrebno je samo pronaći stranice na kojoj se nalaze stavke upita koristeći invertiranu stavka-dokument datoteku te sortirati PageRank vrijednosti tih stranica. PageRank vektor se izračunava samo jednom, do na iznimku dodavanja linkova i stranica.

3.5 Prednosti i mane PageRank metode

Očita prednost PageRank metode je neovisnost rezultata ovisno o korisnikovom upitu. Za vrijeme upita potrebno je samo u invertiranoj stavka-dokument datoteci pronaći relevantne dokumente te ih sortirati po ranije izračunatoj PageRank vrijednosti. Jedan od problema HITS metode je bio *spamming* zbog kojeg je neprestano računata matrica susjedstva te metoda potencija što ovdje očito nije slučaj.

Također, spomenuli smo kako se može manipulirati hub vrijednostima te time i vrijednostima autoriteta stranice dodavanjem linkova. To je slučaj i za PageRank vrijednost, ali taj porast će biti neznatan jer se PageRank vrijednost promatra s obzirom na sve stranice te stoga lokalne promjene poveznica neće imati niti približan utjecaj kao što to imaju u HITS metodi.

Očita prednost je i ta što Google pomoću vektora $\mathbf{v}^T$ kojim definira matricu $\mathbf{E}$ može izravno povećati ili smanjiti PageRank vrijednost određene stranice bez da utječe na jedinstvenost ili brzinu konvergencije metode.

Nedostatak PageRank metode je moguće skretanje s teme (*topic drift*) koje smo također spomenuli kao nedostatak HITS metode. U radu [6] se detaljnije objašnjava taj nedostatak te predlažu novi algoritam *Hilltop* koji na određeni način može prepoznati *experts* stranice čijim poveznicama onda daju veći značaj.

Eventualna mana PageRank metode s obzirom na HITS metodu je ta što HITS metodom dobivamo dva poretka, autoriteta i hub, dok PageRank metodom imamo samo jedan. Također, PageRank metoda je računski znatno složenija, i vremenski i prostorno, od HITS metode.

Poglavlje 4

SALSA

Zadnja metoda za pretraživanje koju ćemo obraditi u ovom radu je SALSA. SALSA (*Stochastic Approach for Link Structure Analysis*) metodu su razvili Lempel i Moran kao svojevrsnu kombinaciju HITS i PageRank metode 2000. godine. Ideja je zadržati dobra svojstva pojedinih modela i kombinirati ih zajedno. Kao što smo opisali, očita prednost HITS metode spram PageRank metode je posjedovanje dva poretka - poredak autoriteta i hub poredak. Također, želimo izbjeći *spamming* koji je značajan problem kod HITS metode. Za početak ćemo na primjeru pokazati rad SALSA metode te zatim usporediti sve tri metode i navesti prednosti odnosno nedostatke pojedine metode spram druge.

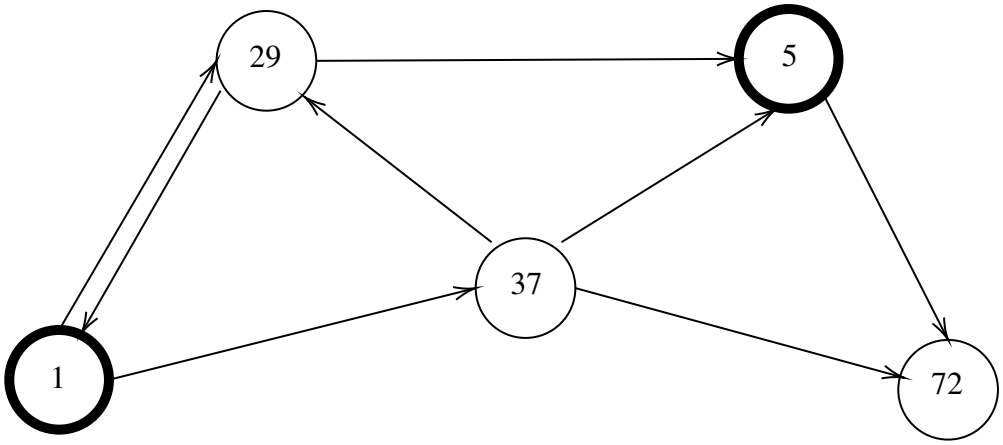
4.1 Primjer SALSA metode

Za početak, Lempel i Moran, su odlučili pratiti HITS metodu. Dakle, pravi se graf susjedstva. Već ovdje vidimo da će očiti nedostatak metode biti ovisnost o korisnikovom upitu. Ipak, nastavimo s daljnim opisom metode. Kako je početak isti kao za HITS metodu uzmimo isti graf susjedstva 2.3 koji smo u drugom poglavlju koristili za primjer. Stavljamo opet sliku grafa 4.1 radi jednostavnosti praćenja daljnjih koraka.

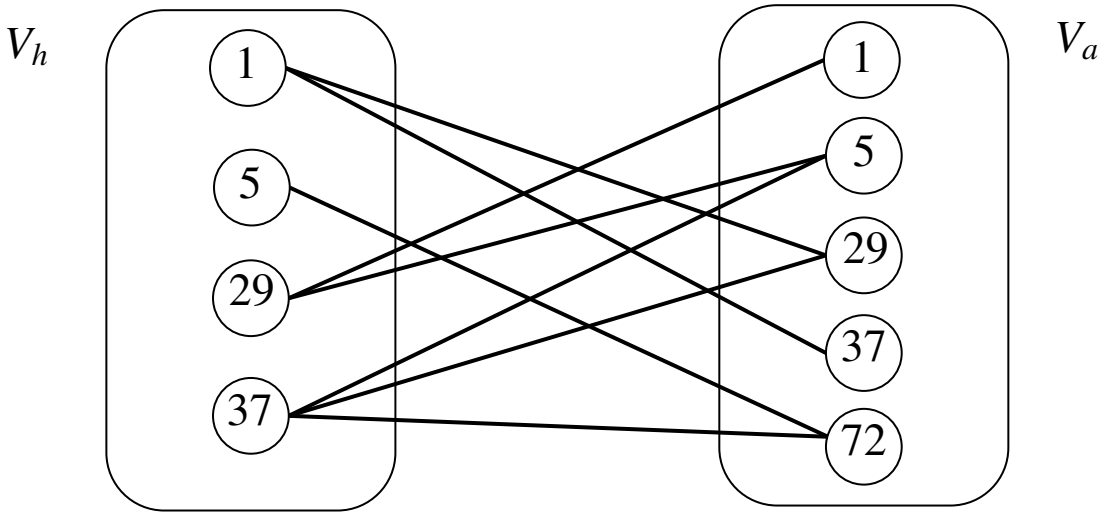
Za razliku od HITS metode ne pravi se matrica susjedstva $\mathbf{L}$. Sljedeći korak SALSA metode je izgradnja bipartitnog neusmjerenog grafa G.

Definicija 4.1.1. *Kažemo da je graf $G = (V, E)$ bipartitan ako se skup vrhova V može podijeliti u dva disjunktna skupa A i B tako da svaki brid iz E povezuje jedan vrh iz A te jedan vrh iz B.*

Graf G izgrađujemo koristeći tri skupa: V_h, V_a i E, gdje V_h označava skup hub čvorova, V_a označava skup čvorova autoriteta te je E skup svih bridova grafa. Dakle, u primjeru 4.1 vidimo da je $V_a = \{1, 5, 29, 37.72\}$ te $V_h = \{1, 5, 29, 37\}$.



Slika 4.1: Graf susjedstva za vrhove 1 i 5



Slika 4.2: Bipartitni graf

Sada graf G dobivamo tako da stavimo vrhove iz V_a i iz V_h na suprotne strane te za svaki usmjereni brid iz E spojimo vrhove neusmjerenim bridom. Na slici 4.2 možemo vidjeti kako izgleda izgrađeni bipartitan graf za naš primjer.

Zatim se grade dva Markovljeva lanca iz grafa G s prijelaznim matricama **H** i **A**, gdje Markovljev lanac s prijelaznom matricom **H** nazivamo hub Markovljev lanac, a Markovljev lanac s prijelaznom matricom **A** nazivamo Markovljev lanaca autoriteta. U svom radu [16] Lempel i Moran daju formulu za računanje matrica **H** i **A**. Ipak do istih matrica može se doći postupkom danim u radu [14] kojeg najvećim dijelom pratimo. Ovim postupkom se

zapravo vidi korištenje dijelova HITS i PageRank metode.

Ne koristeći gotove formule ipak moramo praviti matricu susjedstva $\mathbf{L}$. Prilikom opisa HITS metode dobili smo za ovaj primjer da je matrica susjedstva sljedeća:

$$\mathbf{L} = \begin{array}{c} \begin{matrix} 1 & 5 & 29 & 37 & 72 \end{matrix} \\ \begin{pmatrix} 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array} \begin{matrix} 1 \\ 5 \\ 29 \\ 37 \\ 72 \end{matrix}.$$

Dalje, definiramo matrice $\mathbf{L_r}$ i $\mathbf{L_c}$ koje dobivamo iz matrice $\mathbf{L}$. Matricu $\mathbf{L_r}$ smo dobili normalizacijom redaka matrice $\mathbf{L}$, a matricu $\mathbf{L_c}$ smo dobili normalizacijom stupaca matrice $\mathbf{L}$. Na našem primjeru matrice $\mathbf{L_r}$ i $\mathbf{L_c}$ su sljedeće:

$$\mathbf{L_r} = \begin{array}{c} \begin{matrix} 1 & 5 & 29 & 37 & 72 \end{matrix} \\ \begin{pmatrix} 0 & 0 & 1/2 & 1/2 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1/2 & 1/2 & 0 & 0 & 0 \\ 0 & 1/3 & 1/3 & 0 & 1/3 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array} \begin{matrix} 1 \\ 5 \\ 29 \\ 37 \\ 72 \end{matrix}, \quad \mathbf{L_c} = \begin{array}{c} \begin{matrix} 1 & 5 & 29 & 37 & 72 \end{matrix} \\ \begin{pmatrix} 0 & 0 & 1/2 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1/2 \\ 1 & 1/2 & 0 & 0 & 0 \\ 0 & 1/2 & 1/2 & 0 & 1/2 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array} \begin{matrix} 1 \\ 5 \\ 29 \\ 37 \\ 72 \end{matrix}.$$

Konačno, $\mathbf{H}$ se dobije kao produkt matrica $\mathbf{L_r}\mathbf{L_c}^T$ s izbačenim nul retcima i nul stupcima, a $\mathbf{A}$ se dobije kao produkt matrica $\mathbf{L_c}^T\mathbf{L_r}$ s izbačenim nul retcima i nul stupcima. Stoga, slijedi:

$$\mathbf{L_r}\mathbf{L_c}^T = \begin{array}{c} \begin{matrix} 1 & 5 & 29 & 37 & 72 \end{matrix} \\ \begin{pmatrix} 3/4 & 0 & 0 & 1/4 & 0 \\ 0 & 1/2 & 0 & 1/2 & 0 \\ 0 & 0 & 3/4 & 1/4 & 0 \\ 1/6 & 1/6 & 1/6 & 1/2 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array} \begin{matrix} 1 \\ 5 \\ 29 \\ 37 \\ 72 \end{matrix} \implies \mathbf{H} = \begin{array}{c} \begin{matrix} 1 & 5 & 29 & 37 \end{matrix} \\ \begin{pmatrix} 3/4 & 0 & 0 & 1/4 \\ 0 & 1/2 & 0 & 1/2 \\ 0 & 0 & 3/4 & 1/4 \\ 1/6 & 1/6 & 1/6 & 1/2 \end{pmatrix} \end{array} \begin{matrix} 1 \\ 5 \\ 29 \\ 37 \end{matrix},$$

$$\mathbf{L_c}^T\mathbf{L_r} = \begin{array}{c} \begin{matrix} 1 & 5 & 29 & 37 & 72 \end{matrix} \\ \begin{pmatrix} 1/2 & 1/2 & 0 & 0 & 0 \\ 1/4 & 5/12 & 1/6 & 0 & 1/6 \\ 0 & 1/6 & 5/12 & 1/4 & 1/6 \\ 0 & 0 & 1/2 & 1/2 & 0 \\ 0 & 1/6 & 1/6 & 0 & 2/3 \end{pmatrix} \end{array} \begin{matrix} 1 \\ 5 \\ 29 \\ 37 \\ 72 \end{matrix} \implies \mathbf{A} = \begin{array}{c} \begin{matrix} 1 & 5 & 29 & 37 & 72 \end{matrix} \\ \begin{pmatrix} 1/2 & 1/2 & 0 & 0 & 0 \\ 1/4 & 5/12 & 1/6 & 0 & 1/6 \\ 0 & 1/6 & 5/12 & 1/4 & 1/6 \\ 0 & 0 & 1/2 & 1/2 & 0 \\ 0 & 1/6 & 1/6 & 0 & 2/3 \end{pmatrix} \end{array} \begin{matrix} 1 \\ 5 \\ 29 \\ 37 \\ 72 \end{matrix}.$$

Nadalje, u radu [16] pokazano je da ako je G povezan graf tada su $\mathbf{H}$ i $\mathbf{A}$ ireducibilni Markovljevi lanci te kao što smo već ranije komentirali ireducibilni Markovljevi lanci imaju ireducibilne prijelazne matrice pa će stoga po teoremu 1.2.14 rješenje biti jedinstveno u obliku svojstvenog vektora π_h^T, odnosno π_a^T.

U našem primjeru graf je povezan. Stoga, na isti način kao u PageRank metodi računamo vektore π_h i π_a koji su jedinstveni po Perron Frobenijusovom teoremu. Dobivamo:

$$\begin{matrix} & 1 & 5 & 29 & 37 \end{matrix}$$
$$\pi_h^T = \begin{pmatrix} 1/4 & 1/8 & 1/4 & 3/8 \end{pmatrix},$$

$$\begin{matrix} & 1 & 5 & 29 & 37 & 72 \end{matrix}$$
$$\pi_a^T = \begin{pmatrix} 1/8 & 1/4 & 1/4 & 1/8 & 1/4 \end{pmatrix}.$$

Naravno, postavlja se pitanje računanja vektora ukoliko graf nije povezan. Stoga, pokažimo na primjeru kako pristupiti tom problemu. Pretpostavimo da smo dobili za $\mathbf{A}$ sljedeću matricu:

$$\begin{matrix} & 1 & 5 & 29 & 37 & 72 \end{matrix}$$
$$\mathbf{A} = \begin{pmatrix} 1/2 & 0 & 1/2 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 3/4 & 0 & 1/4 & 0 & 0 \\ 0 & 0 & 0 & 4/5 & 1/5 \\ 0 & 0 & 0 & 1/4 & 3/4 \end{pmatrix} \begin{matrix} 1 \\ 5 \\ 29 \\ 37 \\ 72 \end{matrix}.$$

Iz oblika matrice primjećujemo da $\mathbf{A}$ sadrži tri povezane komponente. Označimo ih s $X = \{1, 29\}$, $Y = \{5\}$ te $Z = \{37, 72\}$. Tada računamo stacionarni vektor posebno za svaku komponentu povezanosti. Dobivamo:

$$\begin{matrix} & 2 \end{matrix}$$
$$\pi_a^T(Y) = \begin{pmatrix} 1 \end{pmatrix}, \quad \begin{matrix} & 1 & 29 \end{matrix}$$
$$\pi_a^T(X) = \begin{pmatrix} 3/5 & 2/5 \end{pmatrix}, \quad \begin{matrix} & 37 & 72 \end{matrix}$$
$$\pi_a^T(Z) = \begin{pmatrix} 5/9 & 4/9 \end{pmatrix}.$$

Kako bi dobili jedan zajednički vektor, u ovakvom slučaju predlaže se skalirati komponente povezanosti s b/n gdje b označava broj vrhova te komponente povezanosti, a n označava sveukupan broj vrhova. Stoga, za ovaj primjer, vektor autoriteta iznosi:

$$\begin{matrix} & 1 & 5 & 29 & 37 & 72 \end{matrix}$$
$$\pi_a^T = \begin{pmatrix} 3/5 \cdot 2/5 & 1 \cdot 1/5 & 2/5 \cdot 2/5 & 5/9 \cdot 2/5 & 4/9 \cdot 2/5 \end{pmatrix}$$

$$\begin{matrix} & 1 & 5 & 29 & 37 & 72 \end{matrix}$$
$$= \begin{pmatrix} 0.24 & 0.2 & 0.16 & 0.22 & 0.18 \end{pmatrix}.$$

Vratimo se sada na naš početni primjer. Dobili smo da vektor autoriteta i hub vektor iznose:

$$\pi_a^T = \begin{matrix}1 & 5 & 29 & 37 & 72\end{matrix}\\ (1/8 \quad 1/4 \quad 1/4 \quad 1/8 \quad 1/4),$$

$$\pi_h^T = \begin{matrix}1 & 5 & 29 & 37\end{matrix}\\ (1/4 \quad 1/8 \quad 1/4 \quad 3/8).$$

Međutim, prilikom računanja HITS metodom dobili smo:

$$\mathbf{x}^T = \begin{matrix}1 & 5 & 29 & 37 & 72\end{matrix}\\ (0.088247 \quad 0.283654 \quad 0.283654 \quad 0.088247 \quad 0.256198),$$

$$\mathbf{y}^T = \begin{matrix}1 & 5 & 29 & 37 & 72\end{matrix}\\ (0.2039 \quad 0.1405 \quad 0.2039 \quad 0.4516 \quad 0).$$

gdje **x** označava vektor autoriteta, a **y** označava hub vektor. Kao što vidimo nismo dobili jednake rezultate, ali poredak je gotovo identičan. Napominjemo da to nije općenito slučaj. HITS metoda i SALSA metoda mogu dati potpuno različite poretke.

4.2 Prednosti i mane SALSA metode

Primijetimo za početak da u slučaju mnogo komponenti povezanosti značajno se smanjuje veličina matrica za koje tražimo stacionarne vektore. Ipak, kao što smo objasnili, više komponenti povezanosti je ekvivalentno s reducibilnosti matrice. Stoga, iako se smanjuje red veličine matrica za koje tražimo svojstveni vektor, zbog reducibilnosti nemamo jedinstveno rješenje. Naravno, moguće je forsiranje reducibilnosti načinom opisanim u PageRank metodi.

Jedan od problema HITS metode je ranjivost na *spamming* zbog međusobne ovisnosti hub vrijednosti i vrijednosti autoriteta. Ponavljamo, *spamming*, u slučaju pretraživanja weba, označava manipulaciju pretraživača kako bi stranica bila bolje rangirana. Kao što se može vidjeti u prethodnom primjeru to se ne događa u tolikoj mjeri jer u SALSA metodi nema tolike ovisnosti između hub vrijednosti i vrijednosti autoriteta. Uostalom, u zadnjem poglavlju ćemo na većem primjeru pokazati koliko *spamming* utječe na svaku metodu.

Također, Lempel i Moran prilikom izglaganje SALSA metode [16] pokazuju kako je SALSA metoda otporna na tzv. TKC problem/efekt. TKC (*tightly knit community*) efekt je pojava kada nekoliko usko povezanih stranica dobiju velike vrijednosti prilikom rangiranja iako nisu autoritativne ili se odnose samo na dio teme tj. korisnikovog upita.

Glavni nedostatak SALSA metode spram PageRank metode je već navedena ovisnost rezultata o korisnikovom upitu te nužnost pravljenja dva Markovljeva lanca za svaki upit. Također, nedostatak osnovne SALSA metode je i konvergencija. Kao što smo spomenuli osnovna metoda ne zahtjeva ireducibilnost Markovljevih lanaca te time konačni vektori nisu jedinstveni.

SALSA metoda je zapravo svojevrsno poboljšanje HITS metode koja je otpornija na sve njene nedostatke. Ipak, ne dostiže razinu PageRank metode otpornosti na navedene nedostatke HITS metode. Dakle, jedina očita prednost SALSA metode spram PageRank metode je dobivanje dva različita poretka stranica.

Poglavlje 5

Usporedba metoda i numerički primjer

Za kraj rada testiramo na većem primjeru sve tri metode. Svi programi su pisani u programskom jeziku Octave verzije 7.2.0.. Za početak za sve tri metode je potrebno napraviti matricu susjedstva. Matricu susjedstva veličine 500×500 smo preuzeli sa stranice [2] koja je dobivena koristeći program surfer.m [3] s početnom stranicom http://www.harvard.edu. Za početak navodimo korištene programske kodove za pojedine metode. Napomenimo da bi za veće matrice trebalo paziti mijenja li se vektor prilikom svakog koraka. Ovdje to nije napravljeno jer se radi s manjom matricom i sigurno je 10000 koraka dovoljno za konvergenciju. Potrebno je promatrati koliko se vektor mijenja prilikom računanja jer, u slučaju ogromnih matrica, računanje može jako dugo trajati te stoga želimo prestati s računanjem u istom trenutku kada smo postigli traženu točnost. Sve programe stavljamo na zasebne stranice radi jednostavnosti praćenja.

Ulaz programa 5.1 za HITS metodu je matrica susjedstva označena s `L`, dok izlazi `vektorPIautoritet` i `vektorPIhub` označavaju redom vektor autoriteta i hub vektor.

Zatim, program 5.2 za SALSA metodu također prima matricu označenu s `L` te također vraća `vektorPIautoritet` i `vektorPIhub` koji označavaju vektor autoriteta i hub vektor.

Konačno, program 5.3 za PageRank metodu također prima matricu susjedstva označenu s `L` te broj alpha koji označava α opisan u PageRank metodi 3.2. Izlaz algoritma je PageRank vektor označen s `vektorPI`.

Sada redom navodimo sve algoritme.

```
function [vektorPIautoritet, vektorPIhub] = hits(L)

  output_precision(10);
  L = L - diag(diag(L));
  n = size(L,1);
  L_t = transpose(L);
  vektorPIautoritet = 1/n+zeros(n,1);
  vektorPIhub = 1/n+zeros(n,1);
  autoritetMatrica = L_t*L;
  hubMatrica = L*L_t;

  for c = 1:10000
    vektorPIautoritet = help1*vektorPIautoritet;
    vektorPIautoritet /= sum(vektorPIautoritet);
  end

  for c = 1:10000
    vektorPIhub = hubMatrica*vektorPIhub;
    vektorPIhub /= sum(vektorPIhub);
  end
  vektorPIautoritet = transpose(vektorPIautoritet);
  vektorPIhub = transpose(vektorPIhub);
end;
```

Algoritam 5.1: HITS metoda

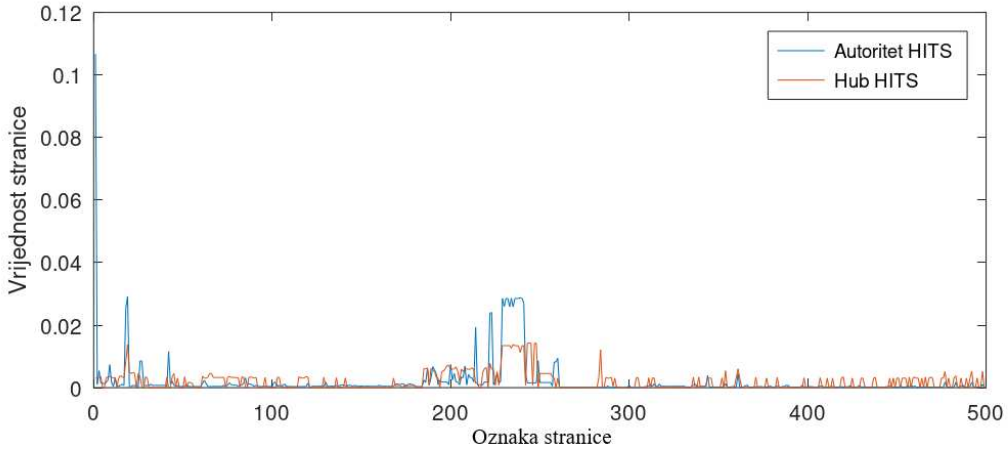
```
function [vektorPIautoritet, vektorPIhub] = salsa(L)

  output_precision(10);
  L = L - diag(diag(L));
  n = size(L,1);
  L_c = L;
  L_r = L;

  for j = 1:n
    if(sum(L_r(j,:))!=0)
      L_r(j,:) /= sum(L_r(j,:));
    end
  end
  for j = 1:n
    if(sum(L_c(:,j))!=0)
      L_c(:,j) /= sum(L_c(:,j));
    end
  end

  H = L_r*transpose(L_c);
  A = transpose(L_c)*L_r;

  vektorPIautoritet = transpose(1/n+zeros(n,1));
  for c = 1:10000
      vektorPIautoritet = vektorPIautoritet*A;
      vektorPIautoritet /= sum(vektorPIautoritet);
  end

  vektorPIhub = transpose(1/n+zeros(n,1));
  for c = 1:10000
      vektorPIhub = vektorPIhub*H;
      vektorPIhub /= sum(vektorPIhub);
  end
end;
```

Algoritam 5.2: SALSA metoda

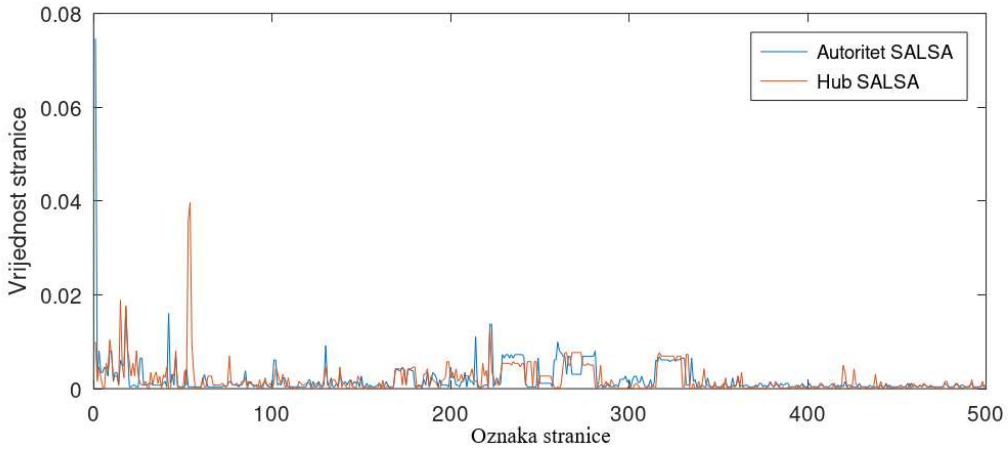
```
function vektorPI = pagerank(L, alpha)

  output_precision(10);
  L = L - diag(diag(L));
  n = size(L,1);
  r = sum(L,2);

  for j = 1:n
    if(r(j) == 0)
      L(j,:) = 1;
     end
  end

  for j = 1:n
    L(j,:) /= sum(L(j,:));
  end

  pomocna = ones(n) / n;

  P = alpha * G + (1-alpha) * pomocna;

  vektorPI = transpose(1/n+zeros(n,1));
  for c = 1:10000
      vektorPI = vektorPI*P;
  end
end;
```

Algoritam 5.3: PageRank metoda

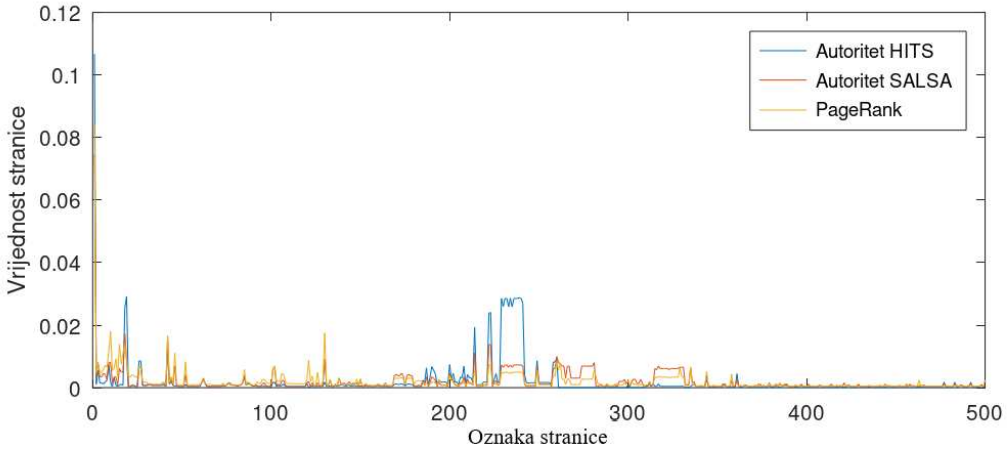
Koristeći navedene programe na sljedećim grafovima možemo vidjeti odnose konačnih rezultata za preuzetu matricu dimenzije 500×500 sa stranice [2]. Na slici 5.1 vidimo odnos vektora autoriteta i hub vektora HITS metode. Na slici 5.2 vidimo odnos vektor autoriteta i hub vektora SALSA metode. Konačno, na slikama 5.3 i 5.4 vidimo odnos vrijednosti konačnih vektora za sve tri metode.



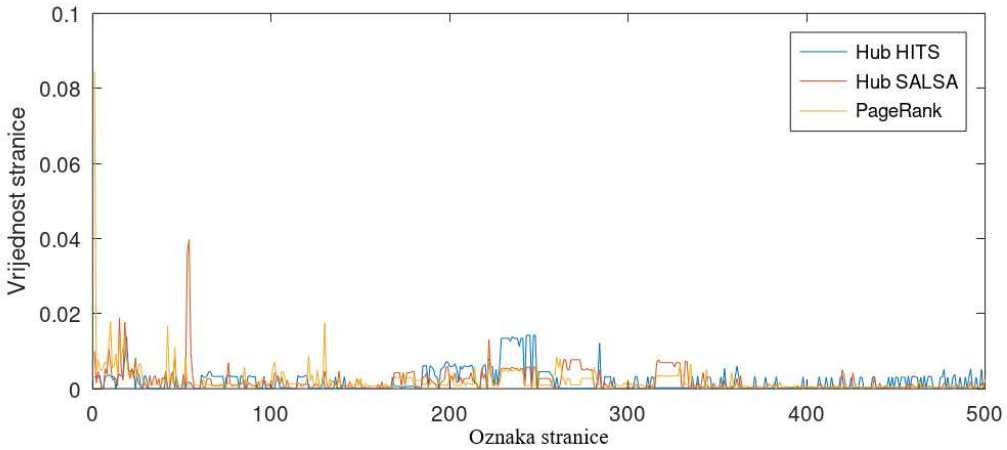
Slika 5.1: Odnos vektora autoriteta i hub vektora HITS metode



Slika 5.2: Odnos vektora autoriteta i hub vektora SALSA metode



Slika 5.3: Odnos vektora autoriteta HITS i SALSA metode te PageRank vektora



Slika 5.4: Odnos hub vektora HITS i SALSA metode te PageRank vektora

U prva dva grafa 5.1 i 5.2 možemo vidjeti kako nema neke značajne povezanosti između vektora autoriteta i hub vektora HITS i SALSA metode. U sljedeća dva grafa 5.3 i 5.4 vidimo kako je PageRank vektor sličan vektorima autoriteta HITS i SALSA metode, a kako nema nikakve očite povezanosti s hub vektorima HITS i SALSA metode. Početna stranica http://www.harvard.edu poprima najveću vrijednost autoriteta za sve tri metode. Razlog tome je način funkcioniranja surfer.m programa. Puno stranica unutar ovih 500 su podstranice početne stranice http://www.harvard.edu. Stoga je vrijednost autoriteta početne

stranice znatno veća od ostalih vrijednosti autoriteta.

Prije nego krenemo pričati o konkretnim rezultatima, navedimo neke osnovne vrijednosti kako bi dobiveni rezultati imali nekakav kontekst. Navedimo medijane dobivenih vektora i standardne devijacije istih. Također, napomenimo kako je prosječna vrijednost svih vektora 0.002. Kao što možemo vidjeti medijan je znatno manji od prosječne vrijed-

	Medijan	**Standardna devijacija**
PageRank	0.0009250691868	0.004369233454
HITS autoritet	0.0004132664415	0.006909651722
HITS hub	0.0002408410883	0.003044473562
SALSA autoritet	0.0007651217596	0.004090013998
SALSA hub	0.0007709214287	0.003370906755

Tablica 5.1: Vrijednosti medijana i standardne devijacije

nosti te je standardna devijacija veća od same prosječne vrijednosti. To označava da je velik broj vrijednosti jako malen te da postoje velike razlike u vrijednostima što se može potvrditi na sva četiri prethodna grafa.

Ranije u radu, posebno u odjeljku 4.2, smo spominjali *spamming*. Probajmo simulirati *spamming* na najjednostavniji način. Napravimo stranicu koja ima poveznice na sve ostale stranice. Koristeći naše programe pogledajmo koje vrijednosti dobivamo za svaku metodu. U tablici 5.2 možemo vidjeti prosječnu apsolutnu grešku između vektora prije *spamming*-a te vektora poslije *spamming*-a.

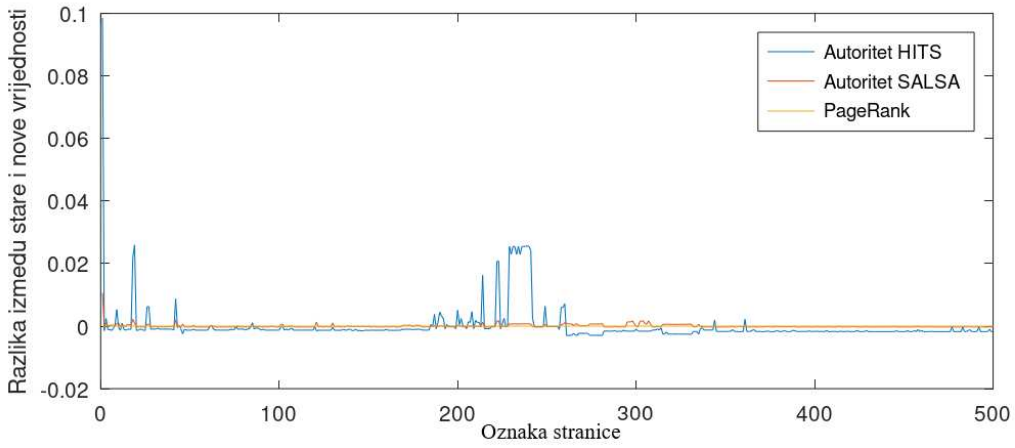
	MAE
PageRank	9.095491087e-07
HITS autoritet	0.002516973370
HITS hub	0.001726221529
SALSA autoritet	0.0003381862808
SALSA hub	0.0003264773098

Tablica 5.2: Prosječna apsolutna greška

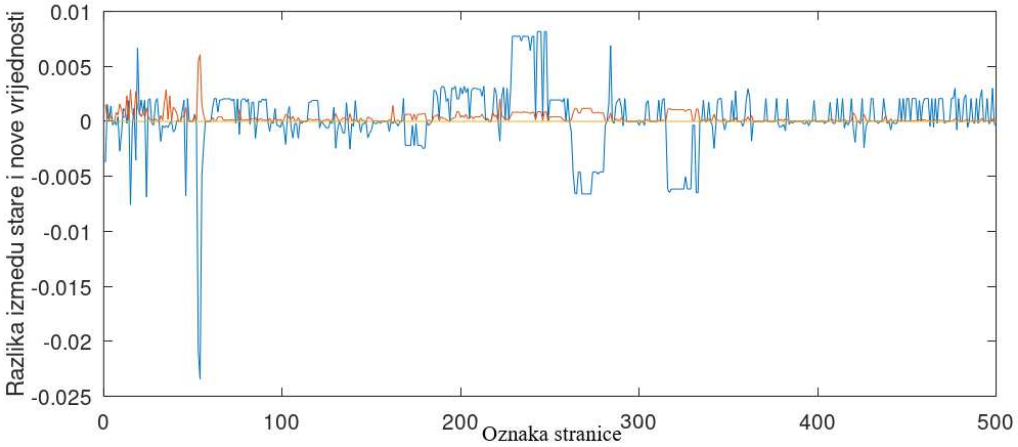
Na slikama 5.5 i 5.6 možemo vidjeti jasnije koliko *spamming* utječe na pojedinu metodu. Vidimo kako SALSA metoda puno bolje podnosi *spamming* od HITS metode.

Ipak, vidi se i koliko je PageRank metoda bolje od obje metode. Razlog tome je jednostavnost manipulacije hub vrijednosti dodavanjem poveznica. Zatim, zbog povezanosti hub vrijednosti i vrijednosti autoriteta dolazi do promjene hub vrijednosti i vrijednosti

autoriteta HITS i SALSA metode. Upravo pokazana otpornost na *spamming* te garantiranost jedinstvenosti rješenja PageRank metode daje joj prednost spram HITS i SALSA metode. Iako, PageRank nema dvojni ranking kao HITS i SALSA metode smatramo kako je značajno bolja i korisnija metoda.



Slika 5.5: Razlika između vektora autoriteta i PageRank vektora prije i poslije *spamming*-a



Slika 5.6: Razlika između hub vektora i PageRank vektora prije i poslije *spamming*-a

Ipak, iako je PageRank metoda puno bolja od HITS i SALSA metode, postoje načini kojima se može utjecati i na vrijednost PageRank-a. Recimo da želimo poboljšati PageRank vrijednost 5 stranica. Recimo da su to stranice koje se nalaze na 100-tom, 200-tom,

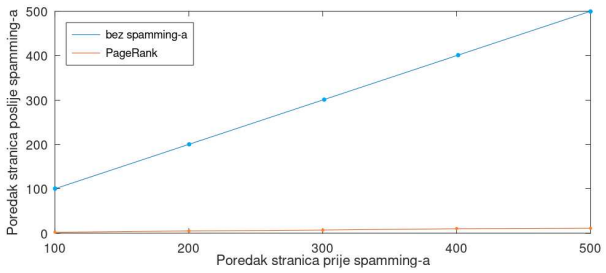
300-tom, 400-tom i 500-tom mjestu poretka. Redom, PageRank vrijednosti i oznake tih stranica su:

$$\begin{aligned}\pi_{277} &= 0.0027365875157\\ \pi_{33} &= 0.0011161635516\\ \pi_{67} &= 0.0008315845146\\ \pi_{371} &= 0.0004727095050\\ \pi_{499} &= 0.0004638236162.\end{aligned}$$

Koristimo ideju *spamming*-a napravljenu u radu [23]. Brišemo sve poveznice sa stranica kojima želimo poboljšati PageRank vrijednost. Zatim, za svaku od tih 5 stranica pravimo 5 novih stranica, te svaka od tih novih 5 stranica ima poveznicu samo na odabranu stranicu kojoj želimo poboljšati PageRank vrijednost. Također, odabranoj stranici pravimo poveznice prema, za nju napravljenih, 5 stranica. Novi rezultati koje dobivamo su:

$$\begin{aligned}\pi_{277} &= 0.02057706288\\ \pi_{33} &= 0.01457467818\\ \pi_{67} &= 0.01320875154\\ \pi_{371} &= 0.01164469779\\ \pi_{499} &= 0.01160396714.\end{aligned}$$

Kao što vidimo naš pokušaj *spamming*-a je iznimno uspješan. Stranice su redom završile na drugom, petom, sedmom, desetom i jedanaestom mjestu u poretku. Rezultate poretka bez *spamming*-a i sa *spamming*-om možemo vidjeti na grafu 5.7. Ipak, postoje algoritmi koji su otporni i na ovakav način *spamming*-a. U radu [23] je naveden DirichtetRank algoritam. Tamo je pokazana njegova otpornost prema ovoj i drugim vrstama *spamming*-a što nije slučaj za PageRank algoritam.



Slika 5.7: Utjecaj *spamming*-a na PageRank

Bibliografija

[1] *How Search Works*, https://www.seroundtable.com/google-130-trillion-pages-22985.html.

[2] *Matrix: MathWorks/Harvard500*, https://www.cise.ufl.edu/research/sparse/matrices/MathWorks/Harvard500.html.

[3] *Using Numerical Computing with MATLAB in the Classroom*, https://www.mathworks.com/matlabcentral/fileexchange/4822-using-numerical-computing-with-matlab-in-the-classroom.

[4] Arvind Arasu, Jasmine Novak, Andrew Tomkins i John Tomlin, *PageRank computation and the structure of the web: Experiments and algorithms*, Proceedings of the Eleventh International World Wide Web Conference, Poster Track, 2002, str. 107–117.

[5] Krishna Bharat i Monika R Henzinger, *Improved algorithms for topic distillation in a hyperlinked environment*, Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval, 1998, str. 104–111.

[6] Krishna Bharat i George A Mihaila, *When experts agree: using non-affiliated experts to rank popular topics*, Proceedings of the 10th international conference on World Wide Web, 2001, str. 597–602.

[7] Chris HQ Ding, Hongyuan Zha, Xiaofeng He, Parry Husbands i Horst D Simon, *Link analysis: hubs and authorities on the World Wide Web*, SIAM review **46** (2004), br. 2, 256–268.

[8] Chris Ding, Xiaofeng He, Parry Husbands, Hongyuan Zha i Horst Simon, *PageRank, HITS and a unified framework for link analysis*, Proceedings of the 2003 SIAM International Conference on Data Mining, SIAM, 2003, str. 249–253.

[9] Zlatko Drmač, *Numerička matematika*, skripta (2010).

[10] Sepandar D Kamvar, Taher H Haveliwala, Christopher D Manning i Gene H Golub, *Extrapolation methods for accelerating PageRank computations*, Proceedings of the 12th international conference on World Wide Web, 2003, str. 261–270.

[11] Sepandar Kamvar, Taher Haveliwala i Gene Golub, *Adaptive methods for the computation of PageRank*, Linear Algebra and its Applications **386** (2004), 51–65.

[12] Sepandar Kamvar, Taher Haveliwala, Christopher Manning i Gene Golub, *Exploiting the block structure of the web for computing pagerank*, Teh. izv., Stanford, 2003.

[13] Kimon P Kontovasilis i Nikolas M Mitrou, *Markov-modulated traffic with nearly complete decomposability characteristics and associated fluid queueing models*, Advances in applied probability **27** (1995), br. 4, 1144–1185.

[14] Amy N Langville i Carl D Meyer, *A survey of eigenvector methods for web information retrieval*, SIAM review **47** (2005), br. 1, 135–161.

[15] Amy N Langville i Carl D Meyer Jr, *Updating the stationary vector of an irreducible Markov chain*, Teh. izv., North Carolina State University. Center for Research in Scientific Computation, 2002.

[16] Ronny Lempel i Shlomo Moran, *The stochastic approach for link-structure analysis (SALSA) and the TKC effect*, Computer Networks **33** (2000), br. 1-6, 387–401.

[17] Carl D Meyer, *Matrix analysis and applied linear algebra*, sv. 71, Siam, 2000.

[18] Mirko Muić, Goran i Primc, *Vektorski prostori*, skripta.

[19] Athanasios N Nikolakopoulos i John D Garofalakis, *NCDawareRank: a novel ranking method that exploits the decomposable structure of the web*, Proceedings of the sixth ACM international conference on Web search and data mining, 2013, str. 143–152.

[20] Davood Rafiei i Alberto O Mendelzon, *What is this page known for? Computing web page reputations*, Computer Networks **33** (2000), br. 1-6, 823–835.

[21] William J Stewart, *Introduction to the numerical solution of Markov chains*, Princeton University Press, 1994.

[22] Zoran Vondraček, *Markovljevi lanci*, skripta (2008).

[23] Xuanhui Wang, Tao Tao, Jian Tao Sun i ChengXiang Zhai, *DirichletRank: Ranking Web Pages Against Link Spams*, Teh. izv., 2005.

[24] Dell Zhang i Yisheng Dong, *An efficient algorithm to rank web resources*, Computer Networks **33** (2000), br. 1-6, 449–455.

[25] Yuhao Zhou, Ruijie Wang, Yi Cheng Zhang, An Zeng i Matúš Medo, *Improving PageRank using sports results modeling*, Knowledge-Based Systems **241** (2022), 108168.

Sažetak

Ukratko, u ovom radu se bavimo metodama za pretraživanje weba. Obrađene metode za pretraživanje weba su bazirane na pronalasku svojstvenog vektora matrice.

U prvom poglavlju obrađena je metoda potencija koja je ključna za sve tri metode za pretraživanje weba. Također, u prvom poglavlju se govori o Perron-Frobenijusovoj teoriji. Glavni rezultat prvog poglavlja je Perron-Frobenijusov teorem koji je potreban za raspravu o jedinstvenosti rješenja metoda za pretraživanje weba.

Svako od sljedeća tri poglavlja predstavlja pojedinu metodu za pretraživanje weba. U drugom poglavlju opisujemo HITS metodu. Metoda HITS definira takozvane hubove (eng. hubs) i autoritete (eng. authorities). Definiramo i pojam matrice susjedstva koju, uz pomoć metode potencija, koristimo za dobivanje poretka stranica prilikom pretraživanja weba.

U trećem poglavlju promatra se PageRank metoda. Metoda PageRank svakoj stranici pridružuje PageRank vrijednost koja mjeri relevantnost neke stranice. PageRank metoda se bazira na ideji je da linkovi s važnijih stranica nose veću težinu od onih s manje važnih stranica. Također, u poglavlju o PageRank metodi prikazan je Markovljev model weba u kojem vidimo direktnu poveznicu između PageRank-ove Google matrice i prijelazne matrice Markovljevog lanca.

U četvrtom poglavlju opisuje se SALSA metoda. SALSA metoda nastoji zadržati prednosti HITS i PageRank metode zajedno te ih kombinirati. SALSA metoda se bazira na izgradnji bipartitnog neusmjerenog grafa iz kojeg se grade dva Markovljeva lanca. Pomoću prijelaznih matrica ta dva Markovljeva lanca i metode potencija dobivamo poredak stranica za traženo pretraživanje weba.

U svakom od poglavlja govori se o konvergenciji metode, prednostima i manama metoda te je za svaku metodu naveden manji primjer.

U zadnjem poglavlju, na većem primjeru, uspoređujemo sve tri metode. Zaključujemo kako je SALSA metoda uspjela popraviti glavne nedostatke HITS metode. Ipak, smatramo da dvojni poredak SALSA metode nije ni približno dovoljna prednost spram PageRank metode koja je bolja u svim ostalim segmentima. Rad završava raspravom o otpornosti metoda na *spamming* u kojoj se vidi osjetljivost HITS i SALSA metode. Rad završavamo s primjerom u kojem vidimo da i PageRank metoda nije otporna na posebne vrste *spamming*-a.

Summary

In this work, we are studying web searching methods. The methods we studied are based on finding the eigenvector of a matrix.

In chapter one we presented the power method, which is the starting point for all the three presented web searching methods. Chapter one also considers Perron-Frobenius theory and gives the Perron-Frobenius theorem which is the main factor in discussing uniqueness of the solution provided by the web searching methods.

The following three chapters present each of the studied methods. Chapter two describes the HITS method which defines so called hubs and authorities. We also defined the adjacency matrix which is used for acquiring page ranking in searching the web, by using the power method.

Chapter three portrays the PageRank method. The method assigns a PageRank score for every page which serves as a measurement of relativity. The basis of the PageRank method is the idea that links with higher relativity pages carry more weigh than those less relevant. This chapter also provides the Markov web model which shows the direct link between the Google matrix's PageRank and the transition matrix of a Markov chain.

Chapter four introduces the SALSA method which tries to hold advantages of both the HITS method and the PageRank method while combining the two. The SALSA method is based on building the bipartite undirected graph which is used for building two Markov chains. Using the transition matrices of the two Markov chains and the power method, we can get the page ranking for the requested searching of the web.

Every chapter provides information about the method's convergence, strengths, and faults, as well as a simple example that displays how does the method work.

The final chapter offers a further example of all the methods combined. We concluded that the SALSA method achieved to fix the main disadvantages of the HITS method. Nevertheless, the dual ranking of the SALSA method is much less enough of an advantage to the PageRank method which exceeds in every other segment. This work concludes in a final discussion of the methods resilience to spamming which shows the HITS method's and the SALSA method's resilience to spamming. The final example of this work shows that the PageRank method isn't resilient to special versions of spamming.

Životopis

Antonio Đurić, rođen je 8. rujna 1998. u Vinkovcima. Osnovnu školu je pohađao u OŠ Ivana Kozarca u Županji. Srednjoškolsko obrazovanje je nastavio u Županji, gdje je završio Prirodslovno-matematičku gimnaziju.

2017. godine upisuje Preddiplomski sveučilišni studij Matematike na Matematičkom odsjeku Prirodoslovno-matematičkog fakulteta Sveučilišta u Zagrebu, koji je 2020. godine i završio. Nakon završenog preddiplomskog studija, upisuje Diplomski sveučilišni studij Računarstva i matematike na istom fakultetu.