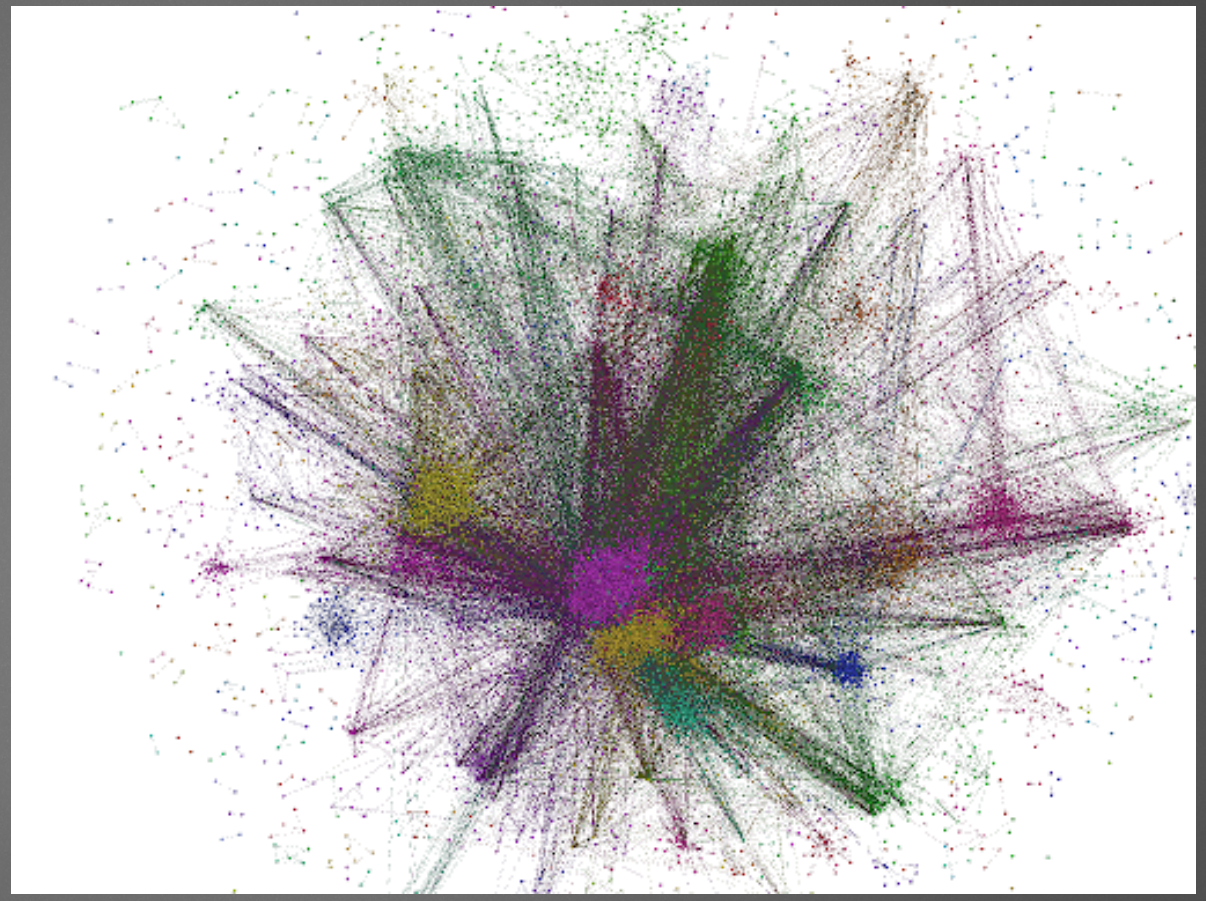
Measuring impact of PageRank in Pop Music

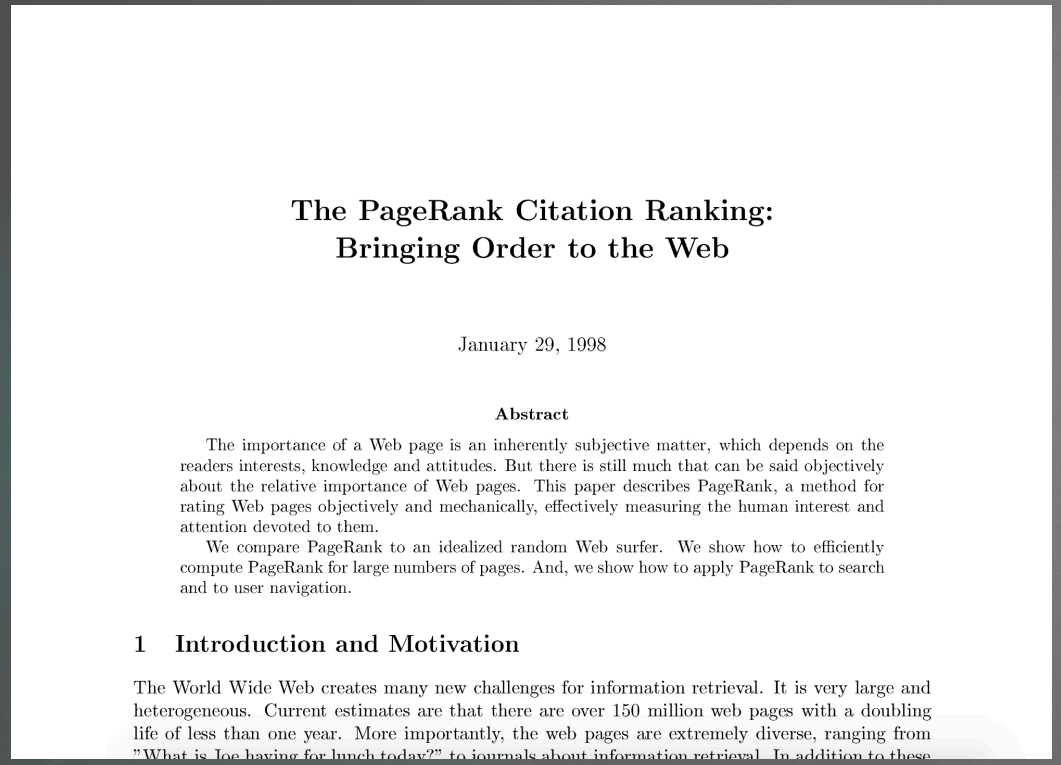
FLAVIA MONTI, 1632488

Web graph

- Nodes = pages
- Edges = links between pages



History



The PageRank Citation Ranking: Bringing Order to the Web

January 29, 1998

Abstract

The importance of a Web page is an inherently subjective matter, which depends on the readers interests, knowledge and attitudes. But there is still much that can be said objectively about the relative importance of Web pages. This paper describes PageRank, a method for rating Web pages objectively and mechanically, effectively measuring the human interest and attention devoted to them.

We compare PageRank to an idealized random Web surfer. We show how to efficiently compute PageRank for large numbers of pages. And, we show how to apply PageRank to search and to user navigation.

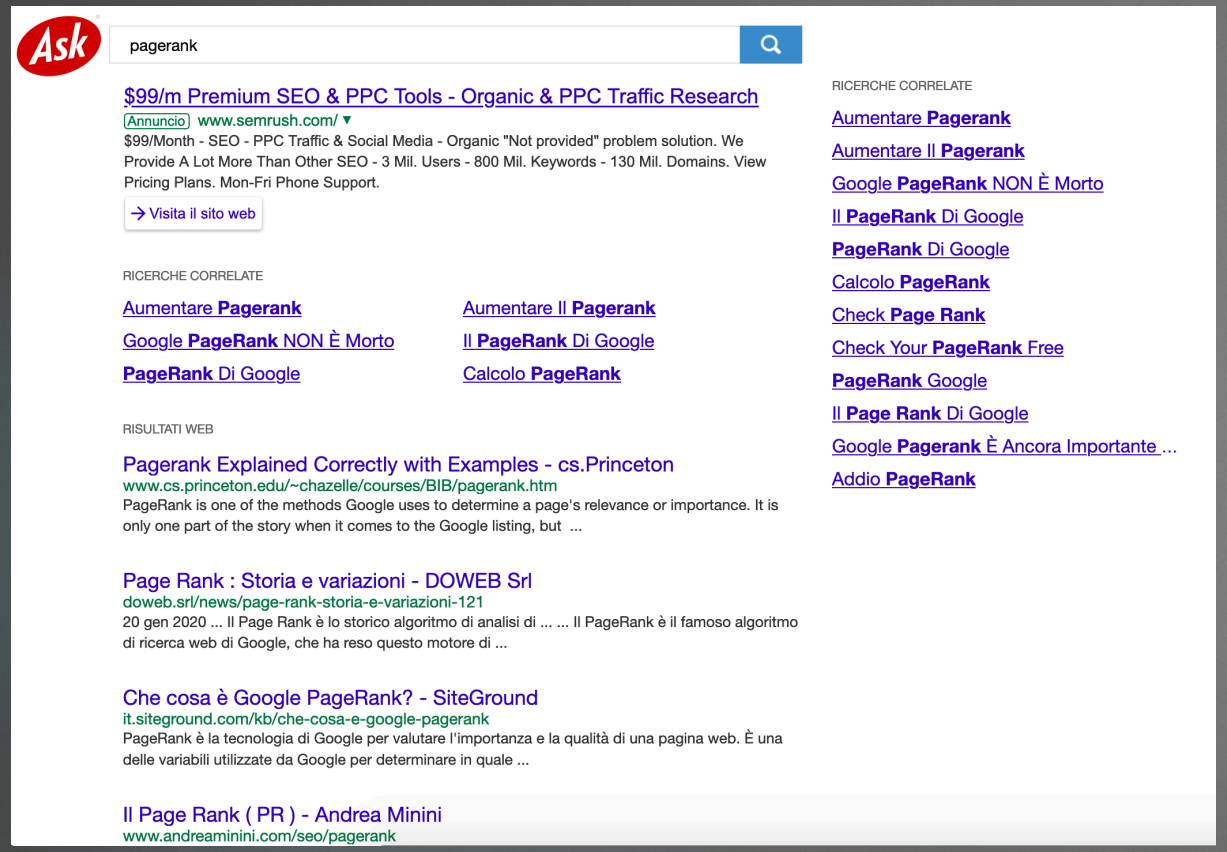
1 Introduction and Motivation

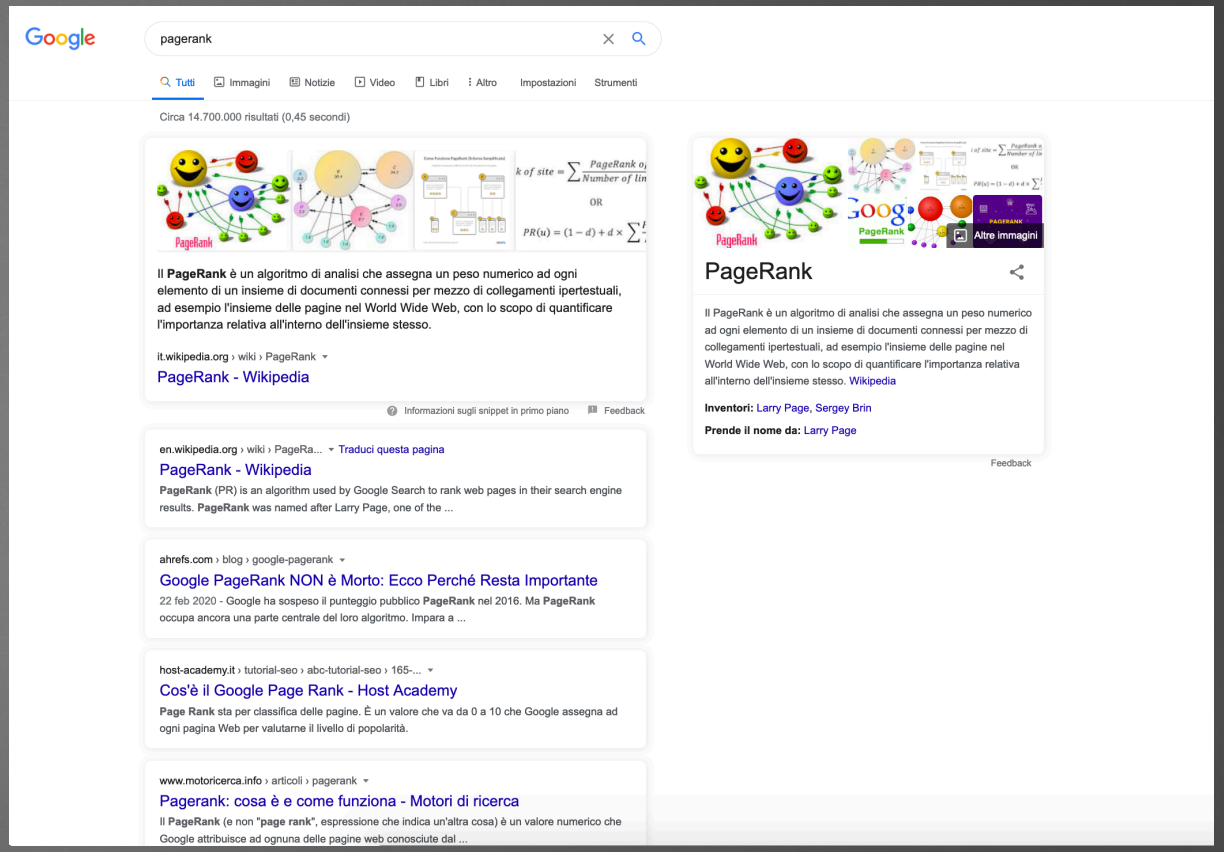
The World Wide Web creates many new challenges for information retrieval. It is very large and heterogeneous. Current estimates are that there are over 150 million web pages with a doubling life of less than one year. More importantly, the web pages are extremely diverse, ranging from "What is Joe having for lunch today?" to journals about information retrieval. In addition to these

In 1998 Sergey Brin and Lawrence Page published their paper presenting the ranking algorithm of the search engine Google, the PageRank.

Thesis: A page is important if it is pointed to by other important pages.

History (2)


pagerank
$99/m Premium SEO & PPC Tools - Organic & PPC Traffic Research
Annuncio www.semrush.com/
$99/Month - SEO - PPC Traffic & Social Media - Organic "Not provided" problem solution. We Provide A Lot More Than Other SEO - 3 Mil. Users - 800 Mil. Keywords - 130 Mil. Domains. View Pricing Plans. Mon-Fri Phone Support.
Visita il sito web
RICERCHE CORRELATE
Aumentare Pagerank
Aumentare Il Pagerank
Google PageRank NON È Morto
Il PageRank Di Google
PageRank Di Google
Calcolo PageRank
RISULTATI WEB
Pagerank Explained Correctly with Examples - cs.Princeton
www.cs.princeton.edu/~chazelle/courses/BIB/pagerank.htm
PageRank is one of the methods Google uses to determine a page's relevance or importance. It is only one part of the story when it comes to the Google listing, but ...
Page Rank : Storia e variazioni - DOWEB Srl
doweb.srl/news/page-rank-storia-e-variazioni-121
20 gen 2020 ... Il Page Rank è lo storico algoritmo di analisi di ... ... Il PageRank è il famoso algoritmo di ricerca web di Google, che ha reso questo motore di ...
Che cosa è Google PageRank? - SiteGround
it.siteground.com/kb/che-cosa-e-google-pagerank
PageRank è la tecnologia di Google per valutare l'importanza e la qualità di una pagina web. È una delle variabili utilizzate da Google per determinare in quale ...
Il Page Rank (PR) - Andrea Minini
www.andreaminini.com/seo/pagerank
RICERCHE CORRELATE
Aumentare Pagerank
Aumentare Il Pagerank
Google PageRank NON È Morto
Il PageRank Di Google
PageRank Di Google
Calcolo PageRank
Check Page Rank
Check Your PageRank Free
PageRank Google
Il Page Rank Di Google
Google Pagerank È Ancora Importante ...
Addio PageRank


Google
pagerank
Tutti
Immagini
Notizie
Video
Libri
Altro
Impostazioni
Strumenti
Circa 14.700.000 risultati (0,45 secondi)
Il PageRank è un algoritmo di analisi che assegna un peso numerico ad ogni elemento di un insieme di documenti connessi per mezzo di collegamenti ipertestuali, ad esempio l'insieme delle pagine nel World Wide Web, con lo scopo di quantificare l'importanza relativa all'interno dell'insieme stesso.
it.wikipedia.org › wiki › PageRank
PageRank - Wikipedia
Informazioni sugli snippet in primo piano
Feedback
en.wikipedia.org › wiki › PageRa... Traduci questa pagina
PageRank - Wikipedia
PageRank (PR) is an algorithm used by Google Search to rank web pages in their search engine results. PageRank was named after Larry Page, one of the ...
ahrefs.com › blog › google-pagerank
Google PageRank NON è Morto: Ecco Perché Resta Importante
22 feb 2020 - Google ha sospeso il punteggio pubblico PageRank nel 2016. Ma PageRank occupa ancora una parte centrale del loro algoritmo. Impara a ...
host-academy.it › tutorial-seo › abc-tutorial-seo › 165-...
Cos'è il Google Page Rank - Host Academy
Page Rank sta per classifica delle pagine. È un valore che va da 0 a 10 che Google assegna ad ogni pagina Web per valutarne il livello di popolarità.
www.motoricerca.info › articoli › pagerank
Pagerank: cosa è e come funziona - Motori di ricerca
Il PageRank (e non "page rank", espressione che indica un'altra cosa) è un valore numerico che Google attribuisce ad ognuna delle pagine web conosciute dal ...
Altre immagini
PageRank
Il PageRank è un algoritmo di analisi che assegna un peso numerico ad ogni elemento di un insieme di documenti connessi per mezzo di collegamenti ipertestuali, ad esempio l'insieme delle pagine nel World Wide Web, con lo scopo di quantificare l'importanza relativa all'interno dell'insieme stesso. Wikipedia
Inventori: Larry Page, Sergey Brin
Prende il nome da: Larry Page
Feedback

PageRank

Initial definition:

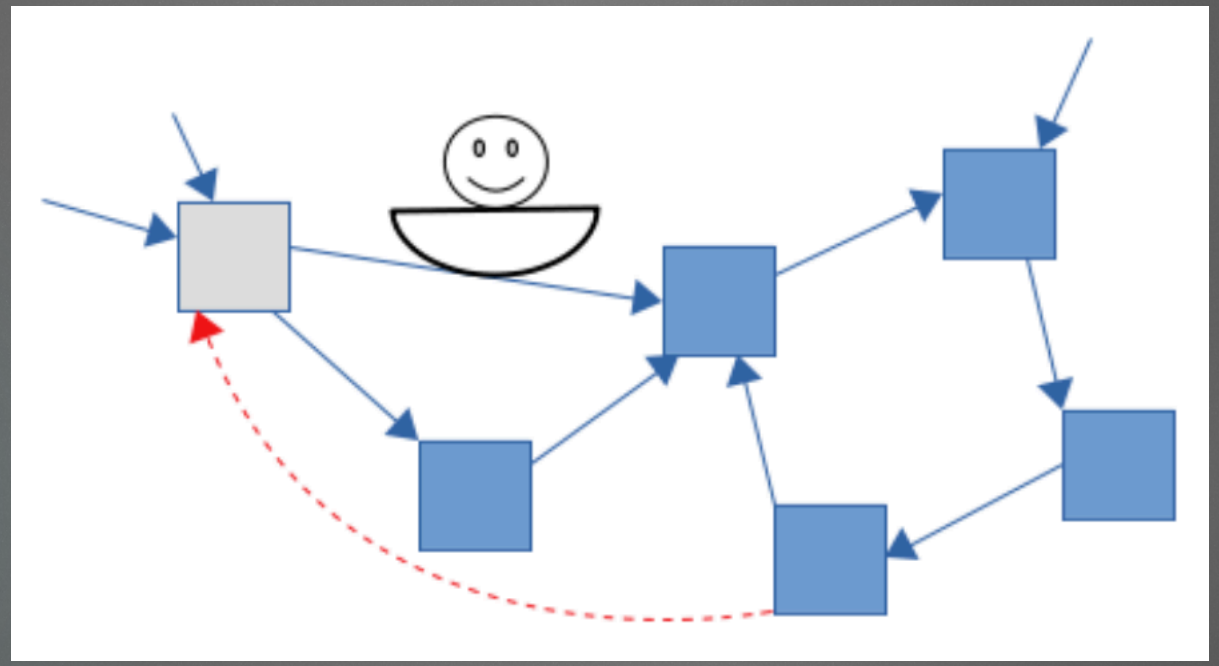
- PageRank R(u), $R(u) = c \sum_{v \in B_u} \frac{R(v)}{|N_v|}$

Another representation

- Matrix H, $H_{uv} = \begin{cases} \frac{1}{|N_u|} \; if \; (u,v) \in E \\ o \; otherwise \end{cases}$
- PageRank vector π^T, $\pi^T_{k+1} = \pi^T_k H$

Random Surfer Model

- A random surfer represent the behaviour of a user navigating on the web, he clicks on successive links at random.



- Dangling nodes problem
- Rational surfer

Modification to the method

- $R = H + a\left(\frac{1}{n}e^T\right),$

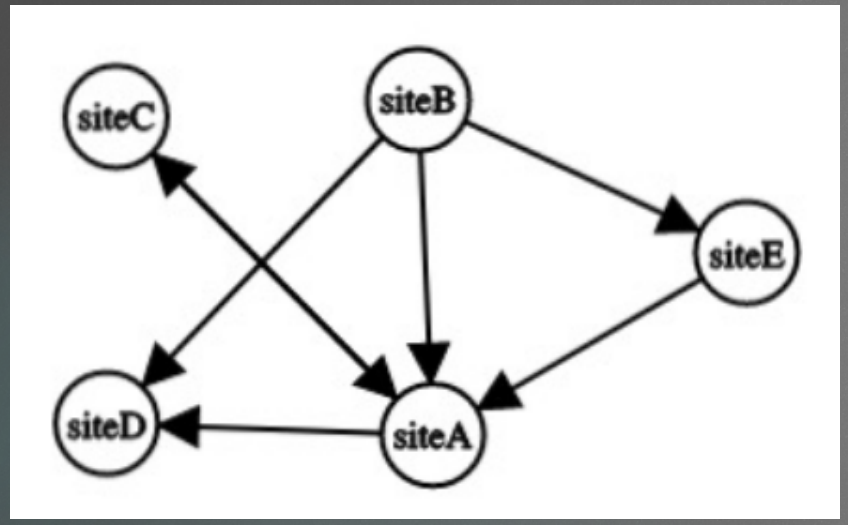
  $a =$ dangling nodes vector

- $R^* = \alpha R + (1 - \alpha)\frac{1}{n}ee^T = \alpha H + \left(\alpha a + (1-\alpha)e\right)\frac{1}{n}e^T$

  R^* is a stochastic primitive matrix

- $\pi_{k+1}^T = \pi_k^T R^*$

Example



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

$$R = \begin{pmatrix} 0 & 0 & 1/2 & 1/2 & 0 \\ 1/3 & 0 & 0 & 1/3 & 1/3 \\ 1 & 0 & 0 & 0 & 0 \\ 1/5 & 1/5 & 1/5 & 1/5 & 1/5 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$R^* = \begin{pmatrix} 3/10 & 3/10 & 17/40 & 17/40 & 3/10 \\ 7/12 & 3/10 & 3/10 & 7/12 & 7/12 \\ 23/20 & 3/10 & 3/10 & 3/10 & 3/10 \\ 3/10 & 3/10 & 3/10 & 3/10 & 3/10 \\ 23/20 & 3/10 & 3/10 & 3/10 & 3/10 \end{pmatrix}$$

Computation

Eigenvector problem

$$\begin{cases} \pi^T = \pi^T R^* \\ \pi^T e = 1 \end{cases}$$

Power method

$\pi_0^T =$ initial_vector;

do

$$\pi_{k+1}^T = \pi_k^T R^*;$$

$$\delta = \left|\left| \pi_{k+1}^T - \pi_k^T \right|\right|_1;$$

while $\delta > \tau$

Recall:

$$\pi_{k+1}^T = \alpha \pi_k^T H + (\alpha \pi_k^T a + 1 - \alpha)\frac{1}{n} e^T$$

$$\pi^T e = 1$$

Parameters of PageRank (1)

Alpha

$\alpha = 0.85, \tau = 10^{-8}$ → number of iterations = 114

$\alpha = 0.99, \tau = 10^{-8}$ → number of iterations = 1833

Random surfer model with $\alpha = 0.85$:

pr = 1/6 → jump to another page

pr = 5/6 → following hyperlink structure of the web

Parameters of PageRank (2)

Uniform probability

$$\frac{1}{n} e e^T$$

query-independent
user-independent

vs

Personalized vector

$$e v^T$$

query-dependent
user-dependent

Convergence and number of iterations

- Rate of convergence of the power method applied to a matrix depends on the ratio of the subdominant and dominant eigenvalues.
- λ_k eigenvalues of R^* and μ_k eigenvalues of H s.t. $\lambda_k = \alpha\mu_k$. Given the structure of the graph, $|\mu_2| = 1 \rightarrow \quad |\lambda_2| = \alpha.$

  Asymptotic rate of convergence is the rate at which $\alpha^k \rightarrow 0$
- A random walk on graph is rapidly-mixing iff the graph is an expander (it has eigenvalues separation).
- Tolerance level measured from the residual $\left|\left|\pi_{k+1}^T - \pi_k^T\right|\right|_1$

  and number of iterations → $\dfrac{\log_{10}\tau}{\log_{10}\alpha}$

Storage

PageRank → $\pi_{k+1}^T = \alpha \pi_k^T H + \left(\alpha \pi_k^T a + 1 - \alpha \right) \frac{1}{n} e^T$

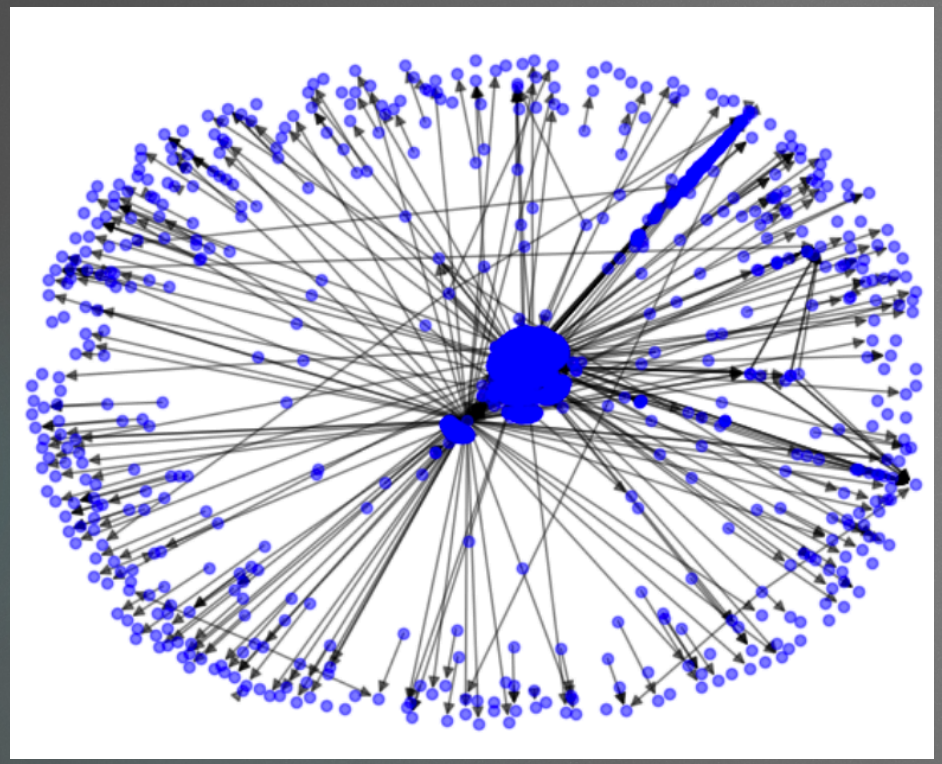
- H = sparse matrix

Requires: $nnz(H)\ multiplications$

- $H = D^{-1} A$

Requires: $n\ multiplications + nnz(H)\ additions$

Pop Music



- Find the most covered band/artist, Covers-graph
- Rank artists depending on collaborations had with other artists, Collaborations-graph

Data

- /covers folder containing info about covers of songs
- .csv files
  - Queen's covers
  - Beatles's covers
  - Bob Dylan's covers
  - …
  - Covers done by Stevie Wonder
  - Most covered artists
  - …

Implementation

$$\pi_{k+1}^T = \alpha \pi_k^T H + \left(\alpha \pi_k^T a + 1 - \alpha\right)\frac{1}{n} e^T$$

```
def pagerank(g, max_iter, alpha, tau):
  sg = nx.stochastic_graph(g)     #stochastic graph
  n_nodes = nx.number_of_nodes(g)
  nodes = g.nodes()

  PI = [1.0/n_nodes] * n_nodes    #initialization of pagerank

  a = []                          #dangling nodes vector
  for n in nodes:
    if g.out_degree(n):
      a.append(1)
    else:
      a.append(0)

  H = nx.adjacency_matrix(sg)     #stochastic matrix
  for i in range(max_iter):
    pi_previous = PI

    v1 = [0] * n_nodes           #v1 = alpha(pi_previous^T*H)
    for r in range(n_nodes):
      row = H[r,:].toarray()
      for c in range(n_nodes):
        v1[c] += pi_previous[c]*row[0][c]
    v1 = [alpha*v for v in v1]

    dang_pi = 0                  #v2 = alpha(pi_previous^T*a)1/n*e^T
    for e in range(n_nodes):
      dang_pi += pi_previous[e]*a[e]
    constant = alpha*dang_pi+1-alpha
    v2 = [float(constant)/n_nodes] * n_nodes

    for e in range(n_nodes):     #pi = v2 + v3
      PI[e] = v1[e] + v2[e]

    PI = normalize(PI)           #pi^T*e=1

    delta = 0                    #check convergence
    for e in range(n_nodes):
      delta += abs(PI[e] - pi_previous[e])
    if delta < tau*n_nodes:
      return transform_pagerank(PI)
  return transform_pagerank(PI)
```

Results

PageRank		PageRank by `networkx`	
Artist	PageRank	Artist	PageRank
BEATLES	0.131089	BEATLES	0.174324
BOB DYLAN	0.076884	BOB DYLAN	0.058624
U2	0.037365	MICHAEL JACKSON	0.026740
MICHAEL JACKSON	0.034016	U2	0.018703
LED ZEPPELIN	0.032358	LED ZEPPELIN	0.016141
BEACH BOYS	0.026401	BEACH BOYS	0.013697
MADONNA	0.018442	BYRON G. HARLAN	0.012248
MISFIT	0.017428	MADONNA	0.009296
QUEEN	0.008226	JOHN LENNON	0.008761
PAUL MCCARTNEY	0.002501	MISFIT	0.008707

$\alpha = 0.85$, $\tau = 10^{-6}$, $\pi_0^T = \frac{1}{n}e^T$, uniform probability $\frac{1}{n}ee^T$

Conclusion

- Vital role in search engine
- Retrieve useful information from the structure of the network