Page Rank

Pia Haberland

Universität Rostock

29.10.2024

Inhaltsverzeichnis

Hintergrund und Bedeutung des PageRank-Algorithmus

- entwickelt von den Gründern von Google
- dient der Bewertung von Webseiten
- misst Relevanz einer Seite basierend auf eingehenden Links
- diente als Kern von Google's Suchmaschinenalgorithmus

  → 2013 von dem Algorithmus Hummingbird ersetzt

  → PageRank nur noch ein Aspekt, der einbezogen wird

Hintergrund und Bedeutung des PageRank-Algorithmus

- Ziel: relevante Webseiten in den Suchergebnissen höher platzieren
- Grundidee:

  Eine Seite ist umso relevanter, je mehr andere relevante Seiten auf sie verlinken.

Grundlage der Markov-Ketten

- Was ist eine Markov-Kette?

  ... sind mathematische Modelle, die benutzt werden, um die Wahrscheinlichkeit von Zustandssequenzen in stochastischen Prozessen vorherzusagen.

- Eigenschaft: "Gedächtnislosigkeit"

  $\rightarrow$ nächste Zustand nur vom aktuellen Zustand abhängt und nicht von der Sequenz, die zu ihm geführt hat

- Werkzeug in Bereichen der Statistik, Wirtschaftswissenschaften und Informatik, um Verhalten und Trends zu analysieren und vorherzusagen

Grundlage der Markov-Ketten - Anwendung auf das Web und den PageRank-Algorithmus

- Modell für zufällige Zusandsübergänge (z.B. Webseitenbesuche)
- Zustände: Webseiten
- Übergangswahrscheinlichkeit: Wahrscheinlichkeit von einem Zustand zum anderen zu wechseln

Mathematische Grundlagen

Graphentheorie im Kontext des Webs

- Webseiten und ihre Verlinkungen lassen sich als gerichteter Graph darstellen, wobei:

  Knoten (Nodes): jede Webseite als Knoten im Graph dargestellt

  Kanten (Edges): Link von einer Webseite zur anderen

Mathematische Grundlagen

Graphentheorie im Kontext des Webs

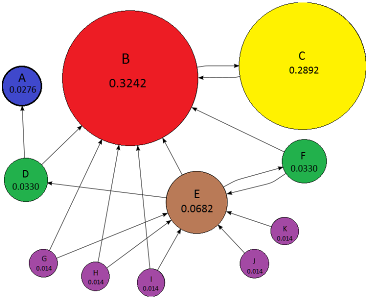

Abbildung: Beispiel PageRank

Mathematische Grundlagen

Übergangsmatrix

- ... beschreibt Wahrscheinlichkeit, von einer Seite zur anderen zu wechseln.
- auch Linkmatrix oder Adjazanzmatrix genannt

Matrixdefinition:

- Sei N die Anzahl an Webseiten im Web.
- Die Übergangsmatrix A ist eine NxN-Matrix, in der der Eintrag

$$a_{i_j}$$

angibt, ob es einen Link von Seite j zu Seite i gibt.

Mathematische Grundlagen

Matrixdefinition:

- Sei N die Anzahl an Webseiten im Web.
- Die Übergangsmatrix A ist eine NxN-Matrix, in der der Eintrag

$$a_{i_j}$$

  angibt, ob es einen Link von Seite j zu Seite i gibt.
- wenn Link existiert, wird Eintrag durch

$$\frac{1}{Anzahl\ ausgehender\ Links\ von\ Seite\ j}$$

- sonst 0

Mathematische Grundlagen

Übergangsmatrix

Beispiel einer einfachen Übergangsmatrix

- 3 Webseiten (A, B und C)
- Seite A hat Links zu B und C, Seite B zu C und Seite C nur zu A.

$$M = \begin{pmatrix} 0 & 0 & 1 \\ 0.5 & 0 & 0 \\ 0.5 & 1 & 0 \end{pmatrix}$$

Mathematische Grundlagen

Stochastische Matrix

- Def.: Eine stochastische Matrix ist eine Matrix, in der jede Spalte zu 1 summiert, da sie die Verteilung der Übergangswahrscheinlichkeiten für jeden Zustand darstellt.
- Notwendigkeit für PageRank:
  damit Algorithmus mathematisch korrekt und stabil läuft, muss die Übergangsmatrix stochastisch sein.
- Jede Seite gibt dabei ihre Relevanz entsprechend der Anzahl und Qualität der Links weiter.
- $\rightarrow$ Normierung der Matrix

Mathematische Grundlagen

Eigenwertproblem und die Berechnung des Page Rank

- PageRank-Algorithmus basiert auf Berechnung des Haupt-Eigenvektors der Übergangsmatrix A
- Eigenwertproblem: Suchen PageRank-Vektor x, der Gleichung erfüllt:

$$Ax = x$$

- Gleichung kann umformuliert werden in Standardform für Eigenwerte:

$$Ax = \lambda x$$

wobei

$$\lambda = 1$$

Mathematische Grundlagen

Eigenwertproblem und die Berechnung des Page Rank

- **Bedeutung für PageRank**: Eigenvektor x repräsentiert die stationäre Verteilung der Besuchswahrschienlichkeit aller Webseiten.
- **Interpretation**: Werte von x geben an, wie oft Seite besucht wird, wenn zufälliger Surfer sehr lange Zeit im Web verbringt
- **Rechenmethoden**: iterative Verfahren

Perron-Frobenius-Satz

- Eine nicht-negative, irreduzible Matrix hat einen eindeutigen größten Eigenwert, der positiv ist. Der Eigenvektor, der diesem Eigenwert entspricht, kann so gewählt werden, dass er ausschließlich positive Einträge enthält.
- Kernpunkte: Existenz eindeutiger positiver Eigenwert, Einfachheit des Eigenwerts (algebraische Vielfachheit $= 1$)
- Sicherstellung einer stabilen Verteilung (PageRank)

Definition des PageRank-Algorithmus

- PageRank-Vektor x ist stationär, sodass

$$Ax = x$$

- Berechnung basiert auf Markov-Kette und zufälligem Surfer-Modell
- Ziel: Relevanz einer Seite anhand ihrer Verlinkung bestimmen

Wie funktioniert der Algorithmus?

- Netzwerk aus Webseiten → Algorithmus → Rangfolge der Webseiten

  1. Erstelle eine **Matrix**, mit der die **Verlinkungen** der Webseiten untereinander beschrieben sind.

[www.youtube.com/watch?v=o0WjbtphaOY]

Wie funktioniert der Algorithmus?

- Netzwerk aus Webseiten → Algorithmus → Rangfolge der Webseiten

  1. Erstelle eine **Matrix**, mit der die **Verlinkungen** der Webseiten untereinander beschrieben sind.

  2. Definiere einen **Vektor**, der so viele **Einträge** wie die **Matrix aus Schritt 1 Spalten** hat.

[www.youtube.com/watch?v=o0WjbtphaOY]

Wie funktioniert der Algorithmus?

- Netzwerk aus Webseiten → Algorithmus → Rangfolge der Webseiten

1. Erstelle eine **Matrix**, mit der die **Verlinkungen** der Webseiten untereinander beschrieben sind.

2. Definiere einen **Vektor**, der so viele **Einträge** wie die **Matrix aus Schritt 1 Spalten** hat.

$$\begin{pmatrix} 1/|Zeilen| \\ ... \\ 1/|Zeilen| \end{pmatrix}$$

[www.youtube.com/watch?v=o0WjbtphaOY]

Wie funktioniert der Algorithmus?

- Netzwerk aus Webseiten $\rightarrow$ Algorithmus $\rightarrow$ Rangfolge der Webseiten

1. Erstelle eine **Matrix**, mit der die **Verlinkungen** der Webseiten untereinander beschrieben sind.

2. Definiere einen **Vektor**, der so viele **Einträge** wie die **Matrix aus Schritt 1 Spalten** hat.

3. **Multipliziere** den Vektor aus **Schritt 2 rechts** an die **Matrix aus Schritt 1**.

[www.youtube.com/watch?v=o0WjbtphaOY]

Wie funktioniert der Algorithmus?

- Netzwerk aus Webseiten → Algorithmus → Rangfolge der Webseiten

1. Erstelle eine **Matrix**, mit der die **Verlinkungen** der Webseiten untereinander beschrieben sind.

2. Definiere einen **Vektor**, der so viele **Einträge** wie die **Matrix aus Schritt 1 Spalten** hat.

3. **Multipliziere** den Vektor aus **Schritt 2 rechts** an die **Matrix aus Schritt 1**.

4. **Wiederhole Schritt 3** mit Vektor, der bei **Schritt 3** als **Ergebnis** herauskommt.

[www.youtube.com/watch?v=o0WjbtphaOY]

Wie funktioniert der Algorithmus?

- Netzwerk aus Webseiten $\rightarrow$ Algorithmus $\rightarrow$ Rangfolge der Webseiten

1. Erstelle eine **Matrix**, mit der die **Verlinkungen** der Webseiten untereinander beschrieben sind.

2. Definiere einen **Vektor**, der so viele **Einträge** wie die **Matrix aus Schritt 1 Spalten** hat.

3. **Multipliziere** den Vektor aus **Schritt 2 rechts** an die **Matrix aus Schritt 1**.

4. **Wiederhole Schritt 3** mit Vektor, der bei **Schritt 3** als **Ergebnis** herauskommt.

5. **Wiederhole Schritt 4** solange, bis im Ergebnisvektor **keine Änderung** mehr auftreten.

[www.youtube.com/watch?v=o0WjbtphaOY]

Beispiel

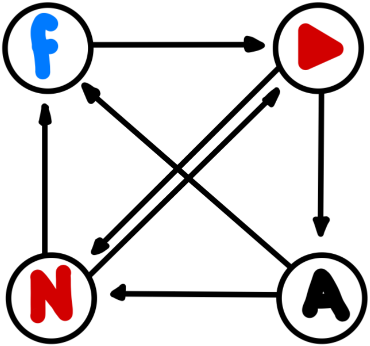

Abbildung: Beispiel - Verlinkung

[www.youtube.com/watch?v=o0WjbtphaOY]

Beispiel

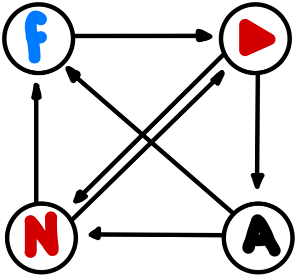

Abbildung: Beispiel - Verlinkung

$$\begin{array}{cccc} f & Y & A & N \\ \end{array}$$
$$\begin{pmatrix} \quad & \quad & \quad & \quad \\ & & & \\ & & & \\ & & & \end{pmatrix}$$

Beispiel

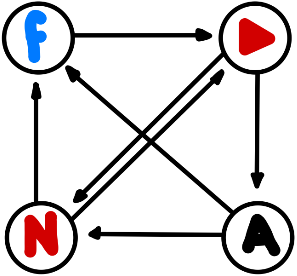

Abbildung: Beispiel - Verlinkung

$$\begin{array}{cccc} f & Y & A & N \\ \multicolumn{4}{c}{\begin{pmatrix} 0 & & & \\ 1 & & & \\ 0 & & & \\ 0 & & & \end{pmatrix}} \end{array}$$

Beispiel

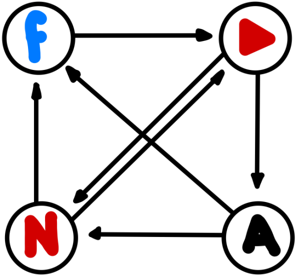

Abbildung: Beispiel - Verlinkung

$$\begin{array}{c}
\begin{array}{cccc} \mathrm{f} & \mathrm{Y} & \mathrm{A} & \mathrm{N} \end{array} \\
\begin{pmatrix} 0 & 0 & & \\ 1 & 0 & & \\ 0 & 0.5 & & \\ 0 & 0.5 & & \end{pmatrix}
\end{array}$$

Beispiel

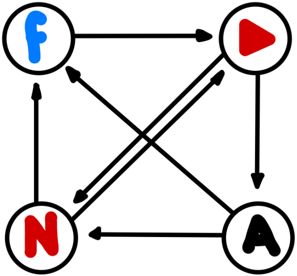

Abbildung: Beispiel - Verlinkung

$$
\begin{array}{c}
\begin{array}{cccc} \text{f} & \text{Y} & \text{A} & \text{N} \end{array} \\
\begin{pmatrix} 0 & 0 & 0.5 & \\ 1 & 0 & 0 & \\ 0 & 0.5 & 0 & \\ 0 & 0.5 & 0.5 & \end{pmatrix}
\end{array}
$$

Beispiel

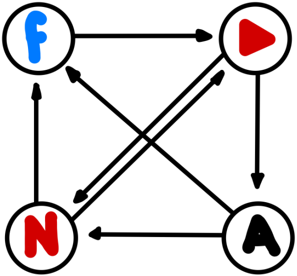

Abbildung: Beispiel - Verlinkung

$$
\begin{array}{c}
\begin{array}{cccc} \text{f} & \text{Y} & \text{A} & \text{N} \end{array} \\
\begin{pmatrix}
0 & 0 & 0.5 & 0.5 \\
1 & 0 & 0 & 0.5 \\
0 & 0.5 & 0 & 0 \\
0 & 0.5 & 0.5 & 0
\end{pmatrix}
\end{array}
$$

Beispiel

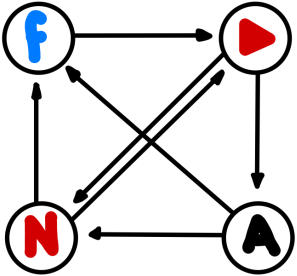

Abbildung: Beispiel - Verlinkung

$$\begin{pmatrix} 0 & 0 & 0.5 & 0.5 \\ 1 & 0 & 0 & 0.5 \\ 0 & 0.5 & 0 & 0 \\ 0 & 0.5 & 0.5 & 0 \end{pmatrix} \times \begin{pmatrix} 0.25 \\ 0.25 \\ 0.25 \\ 0.25 \end{pmatrix}$$

Beispiel

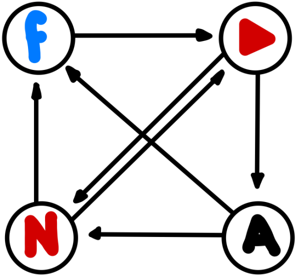

Abbildung: Beispiel - Verlinkung

$$\begin{pmatrix} 0 & 0 & 0.5 & 0.5 \\ 1 & 0 & 0 & 0.5 \\ 0 & 0.5 & 0 & 0 \\ 0 & 0.5 & 0.5 & 0 \end{pmatrix} \times \begin{pmatrix} 0.25 \\ 0.25 \\ 0.25 \\ 0.25 \end{pmatrix} = \begin{pmatrix} 0.25 \\ 0.375 \\ 0.125 \\ 0.25 \end{pmatrix}$$

Beispiel

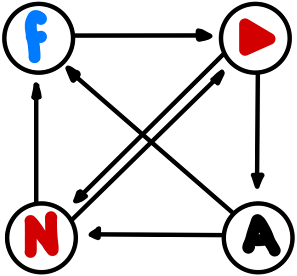

Abbildung: Beispiel - Verlinkung

$$\begin{pmatrix} 0 & 0 & 0.5 & 0.5 \\ 1 & 0 & 0 & 0.5 \\ 0 & 0.5 & 0 & 0 \\ 0 & 0.5 & 0.5 & 0 \end{pmatrix} \times \begin{pmatrix} 0.25 \\ 0.375 \\ 0.125 \\ 0.25 \end{pmatrix} =$$

Beispiel

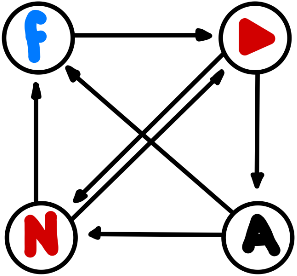

Abbildung: Beispiel - Verlinkung

$$\begin{pmatrix} 0 & 0 & 0.5 & 0.5 \\ 1 & 0 & 0 & 0.5 \\ 0 & 0.5 & 0 & 0 \\ 0 & 0.5 & 0.5 & 0 \end{pmatrix} \times \begin{pmatrix} 0.25 \\ 0375 \\ 0.125 \\ 0.25 \end{pmatrix} = \begin{pmatrix} 0.1875 \\ 0.375 \\ 0.1875 \\ 0.25 \end{pmatrix}$$

Beispiel

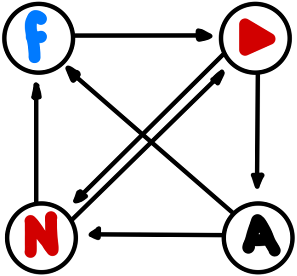

Abbildung: Beispiel - Verlinkung

$$\begin{pmatrix} 0 & 0 & 0.5 & 0.5 \\ 1 & 0 & 0 & 0.5 \\ 0 & 0.5 & 0 & 0 \\ 0 & 0.5 & 0.5 & 0 \end{pmatrix} \times \begin{pmatrix} 0.1875 \\ 0.375 \\ 0.1875 \\ 0.25 \end{pmatrix} = \begin{pmatrix} 0.21875 \\ 0.3125 \\ 0.1875 \\ 0.28125 \end{pmatrix}$$

Beispiel

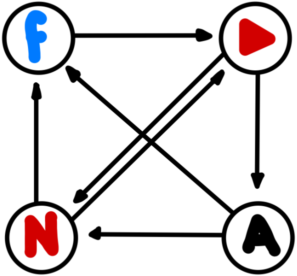

Abbildung: Beispiel - Verlinkung

konvergiert

$$\begin{pmatrix} 0 & 0 & 0.5 & 0.5 \\ 1 & 0 & 0 & 0.5 \\ 0 & 0.5 & 0 & 0 \\ 0 & 0.5 & 0.5 & 0 \end{pmatrix} \times \ldots \approx \begin{pmatrix} 0.217 \\ 0.348 \\ 0.174 \\ 0.261 \end{pmatrix}$$

Formel für PageRank mit Dämpfungsfaktor

- Dämpfungsfaktor p=0.85

  Wahrscheinlichkeit einem Link zu folgen

- zufälliger Sprung

Aufbau der Übergangsmatrix für das Web

- $$A = pGD + (1-p)\frac{1}{n}ez^T \tag{1}$$

A: Übergangsmatrix

p: Dämpfungsfaktor (typischerweise 0.85)

D: Diagonalmatrix

G: Matrix mit Verlinkungsstruktur zwischen Seiten (1 oder 0)

e: Vektor mit nur Einselementen

z^T:

$$z_j = \begin{cases} \delta, & wenn\ c_j \neq 0 \\ 1/n, & wenn\ c_j = 0 \end{cases} .$$

$1/n$: Normierungskonstante mit n Anzahl der Seiten

Aufbau der Übergangsmatrix für das Web

- $$A = pGD + (1-p)\frac{1}{n}ez^T$$

Beispiel:
Angenommen, wir haben ein kleines Netzwerk von 3 Seiten (n=3) und der Dämpfungsfaktor p beträgt 0.85. Die Struktur der Links (G) sei wie folgt:

Seite 1 verlinkt auf Seite 2 und 3
Seite 2 verlinkt auf Seite 3
Seite 3 verlinkt auf Seite 1

Aufbau der Übergangsmatrix für das Web

- $$A = p\, G\, D + (1-p)\, \frac{1}{n}\, e\, z^T$$

Beispiel:

Angenommen, wir haben ein kleines Netzwerk von 3 Seiten (n=3) und der Dämpfungsfaktor p beträgt 0.85. Die Struktur der Links (G) sei wie folgt:

Seite 1 verlinkt auf Seite 2 und 3
Seite 2 verlinkt auf Seite 3
Seite 3 verlinkt auf Seite 1

$$G = \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$$

$$D = \begin{pmatrix} 0.5 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Aufbau der Übergangsmatrix für das Web

- $$A = p\, G\, D + (1-p)\, \frac{1}{n}\, e\, z^T$$

Beispiel:

$$p\, G\, D = 0.85 \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0.5 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 0.85 & 0.85 \\ 0 & 0 & 0.85 \\ 0.425 & 0 & 0 \end{pmatrix}$$

Aufbau der Übergangsmatrix für das Web

- $$A = p\,G\,D + (1-p)\,\frac{1}{n}\,e\,z^T$$

Beispiel:

$$(1-p)\,\frac{1}{n}\,e\,z^T = (1-0.85)\,\frac{1}{3}\begin{pmatrix}1\\1\\1\end{pmatrix}\begin{pmatrix}1 & 1 & 1\end{pmatrix} = 0.05\begin{pmatrix}1 & 1 & 1\\1 & 1 & 1\\1 & 1 & 1\end{pmatrix}$$

Aufbau der Übergangsmatrix für das Web

- $$A = p\; G\; D + (1-p)\,\frac{1}{n}\; e\; z^T$$

Beispiel:

$$(1-p)\,\frac{1}{n}\; e\; z^T = (1-0.85)\,\frac{1}{3}\begin{pmatrix}1\\1\\1\end{pmatrix}\begin{pmatrix}1 & 1 & 1\end{pmatrix} = \begin{pmatrix}0.05 & 0.05 & 0.05\\0.05 & 0.05 & 0.05\\0.05 & 0.05 & 0.05\end{pmatrix}$$

Aufbau der Übergangsmatrix für das Web

- $$A = p\, G\, D + (1-p)\,\frac{1}{n}\, e\, z^T$$

Beispiel:

$$A = \begin{pmatrix} 0 & 0.85 & 0.85 \\ 0 & 0 & 0.85 \\ 0.425 & 0 & 0 \end{pmatrix} + \begin{pmatrix} 0.05 & 0.05 & 0.05 \\ 0.05 & 0.05 & 0.05 \\ 0.05 & 0.05 & 0.05 \end{pmatrix} = \begin{pmatrix} 0.05 & 0.9 & 0.9 \\ 0.05 & 0.05 & 0.9 \\ 0.475 & 0.05 & 0.05 \end{pmatrix}$$

Aufbau der Übergangsmatrix für das Web

Beispiel:
anschließend Normierung

$$\begin{pmatrix} 0.05 & 0.9 & 0.9 \\ 0.05 & 0.05 & 0.9 \\ 0.475 & 0.05 & 0.05 \end{pmatrix}$$

$$\sum 0.575 \sum 1,0 \sum 1.85$$

Aufbau der Übergangsmatrix für das Web

Beispiel:
Spalte 1

$$\sum = 0.575$$

$$A_{11} = \frac{0.05}{0.575} \approx 0.087$$

$$A_{21} = \frac{0.05}{0.575} \approx 0.087$$

$$A_{31} = \frac{0.475}{0.575} \approx 0.826$$

Aufbau der Übergangsmatrix für das Web

Beispiel:
Spalte 2:

$$\sum = 1.0$$

$$A_{12} = 0.9$$

$$A_{22} = 0.05$$

$$A_{33} = 0.05$$

Aufbau der Übergangsmatrix für das Web

Beispiel:
Spalte 3:

$$\sum = 1.85$$

$$A_{13} = \frac{0.9}{1.85} \approx 0.486$$

$$A_{23} = \frac{0.9}{1.85} \approx 0.486$$

$$A_{33} = \frac{0.05}{1.85} \approx 0.027$$

Aufbau der Übergangsmatrix für das Web

Beispiel:

normierte Matrix:

$$\begin{pmatrix} 0.087 & 0.9 & 0.486 \\ 0.087 & 0.05 & 0.486 \\ 0.826 & 0.05 & 0.027 \end{pmatrix}$$

Stationäre Verteilung und Gleichungssystem

- stationärer Vektor x als Lösung

$$x = Ax$$

- eindeutige Lösung, die als PageRank-Vektor verwendet wird

Power-Methode zur Bestimmung Eigenvektors

- iterative Methode zur Bestimmung betragsmäßig größten Eigenwerts und des dazugehörigen Eigenvektors
- gut geeignet für Sparse-Matrizen wie im PageRank
- **Abbruchkriterien**:

  Änderung des Eigenvektors

  $$||x_{k+1} - x_k|| < \epsilon$$

  vorher festgelegt maximale Iterationanzahl

Effizienz und Grenzen der Power-Methode

- Vorteil: effizient für Sparse-Matrizen, speichert nur Nicht-Null-Elemente
- Nachteil: langsame Konvergenz bei schlecht konditionierten Matrizen
- Lösung: optimierte Algorithmen und verteiltes Rechnen

Definition und Eigenschaften von Sparse-Matrizen

- Speicherung nur der Nicht-Null-Elemete (Speicherplatzersparnis)
- Anwendung: Netzwerk- und Graphenanalyse
- Matrixdarstellung des Webs: extrem viele Nullwerte
- Matlab-Funktionen:

  `spdiags, sparse, full, find`

- Sparse-Matrixstruktur spart Speicher und Rechenleistung

Effizienzgewinne durch Sparse-Matrizen

- durch reduzierten Speicherbedarf kürzere Berechnungszeiten
- Vergleich von Speicherbedarf zwischen Sparse- und Vollmatrizen
- Beispiel:

$$A = \begin{pmatrix} 0 & 0 & 0 & 5 \\ 0 & 2 & 0 & 0 \\ 1 & 3 & 0 & 0 \\ 0 & 0 & 4 & 0 \end{pmatrix}$$

```
S = sparse(A)
S = sparse(i,j,s,m,n)
S = sparse([3 2 3 4 1],[1 2 2 3 4],[1 2 3 4 5],4,4)
```

Numerische Probleme - Rundungsfehler und Konvergenz

- Rundungsfehler können Konvergenz verzögern
- besonders problematisch bei schlecht konditionierten Matrizen
- Lösung: Stabilisierung durch Anpassung des Dämpfungsfaktors

Dead Ends

- Seiten, die keine ausgehenden Links haben

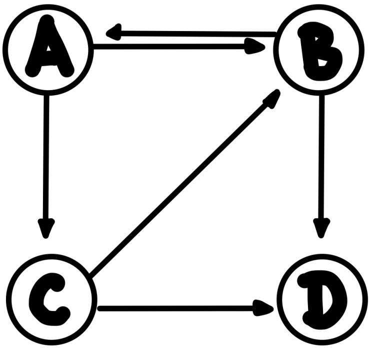

Abbildung: Beispiel - Verlinkung mit Dead End

- Knoten D ist ein Dead End
- Wie würde nun der PageRank Algorithmus reagieren?

Dead Ends

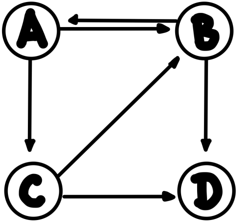

Abbildung: Beispiel - Verlinkung mit Dead End

- Berechnung:

$$\begin{pmatrix} 0 & 0.5 & 0 & 0 \\ 0.5 & 0 & 0.5 & 0 \\ 0.5 & 0 & 0 & 0 \\ 0 & 0.5 & 0.5 & 0 \end{pmatrix}$$

Dead Ends

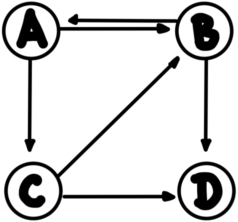

Abbildung: Beispiel - Verlinkung mit Dead End

- Berechnung:

$$\begin{pmatrix} 0 & 0.5 & 0 & 0 \\ 0.5 & 0 & 0.5 & 0 \\ 0.5 & 0 & 0 & 0 \\ 0 & 0.5 & 0.5 & 0 \end{pmatrix} \times \begin{pmatrix} 1/4 \\ 1/4 \\ 1/4 \\ 1/4 \end{pmatrix} =$$

Dead Ends

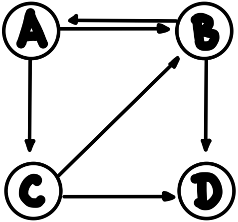

Abbildung: Beispiel - Verlinkung mit Dead End

- Berechnung:

$$\begin{pmatrix} 0 & 0.5 & 0 & 0 \\ 0.5 & 0 & 0.5 & 0 \\ 0.5 & 0 & 0 & 0 \\ 0 & 0.5 & 0.5 & 0 \end{pmatrix} \times \begin{pmatrix} 1/4 \\ 1/4 \\ 1/4 \\ 1/4 \end{pmatrix} = \begin{pmatrix} 1/8 \\ 1/4 \\ 1/8 \\ 1/4 \end{pmatrix}$$

Dead Ends

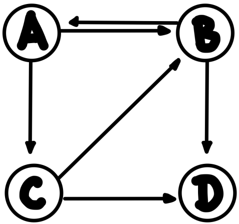

Abbildung: Beispiel - Verlinkung mit Dead End

- Berechnung:

$$\begin{pmatrix} 0 & 0.5 & 0 & 0 \\ 0.5 & 0 & 0.5 & 0 \\ 0.5 & 0 & 0 & 0 \\ 0 & 0.5 & 0.5 & 0 \end{pmatrix} \times \ldots = \begin{pmatrix} 0.0004959 \\ 0.0006561 \\ 0.0003738 \\ 0.0007781 \end{pmatrix}$$

Dead Ends

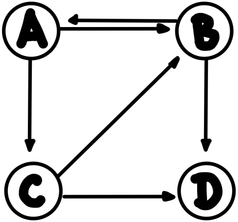

Abbildung: Beispiel - Verlinkung mit Dead End

- Berechnung:

$$\begin{pmatrix} 0 & 0.5 & 0 & 0 \\ 0.5 & 0 & 0.5 & 0 \\ 0.5 & 0 & 0 & 0 \\ 0 & 0.5 & 0.5 & 0 \end{pmatrix} \times \ldots = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Dead Ends

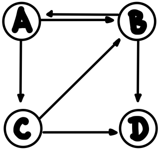

Abbildung: Beispiel - Verlinkung mit Dead End

- Lösung:
- Verhinderung des Hängenbleibens durch sogenannte Teleports
- Wahrscheinlichkeit, dass auf bestimmter Seite gesurft wird

$$\frac{1}{n}$$

- n: Anzahl Seiten

Spider Traps

- C und D sind eine Spider Trap

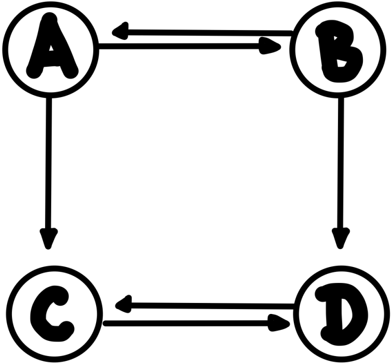

Abbildung: Beispiel - Verlinkung mit Spider Trap

Effizienzprobleme bei sehr großen Matrizen

- Berechnung bei Milliarden Webseiten speicherintensiv und komplex
- Notwendigkeit von verteiltem Rechnen für PageRank-Berechnungen

Manipulation dur SEO

- Linkfarmen und andere Strategien können den PageRank manipulieren
- **Gegenmaßnahme**: Google überwacht Verlinkungsmuster und passt den Algorithmus regelmäßig an, um Manipulation zu erschweren

Quellen

- Moler, C. B. (2010). Numerical Computing with MATLAB: Revised Reprint. SIAM.
- Varga, R. S. (2000). Matrix Iterative Analysis. Springer.
- Brin, S., u. Page, L. (1998). The PageRank Citation Ranking: Bringing Order to the Web. Stanford InfoLab.
- Langville, A. N., u. Meyer, C. D. (2006). Google's PageRank and Beyond: The Science of Search Engine Rankings. Princeton University Press.

- www.youtube.com/watch?v=o0WjbtphaOY
- https://www.suchmaschinen-doktor.de/algorithmen/pagerank.html

Bildquellen

- Abbildung 1: Beispiel PageRank https://sebastianviereck.de/pagerank-algorithmus-java/
- Abbildung 2: Beispiel Verlinkung - selbst erstellt mit Goodnotes
- Abbildung 3: Beispiel - Verlinkung mit Dead End - selbst erstellt mit Goodnotes
- Abbildung 4: Beispiel - Verlinkung mit Spider Trap - selbst erstellt mit Goodnotes