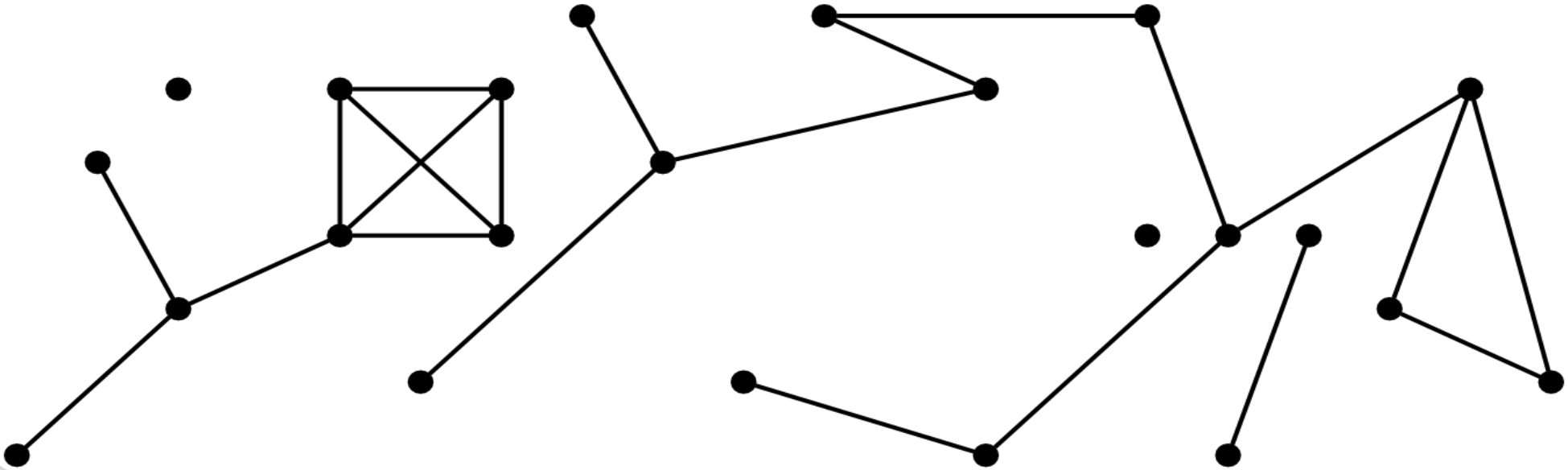
Karlsruher Institut für Technologie

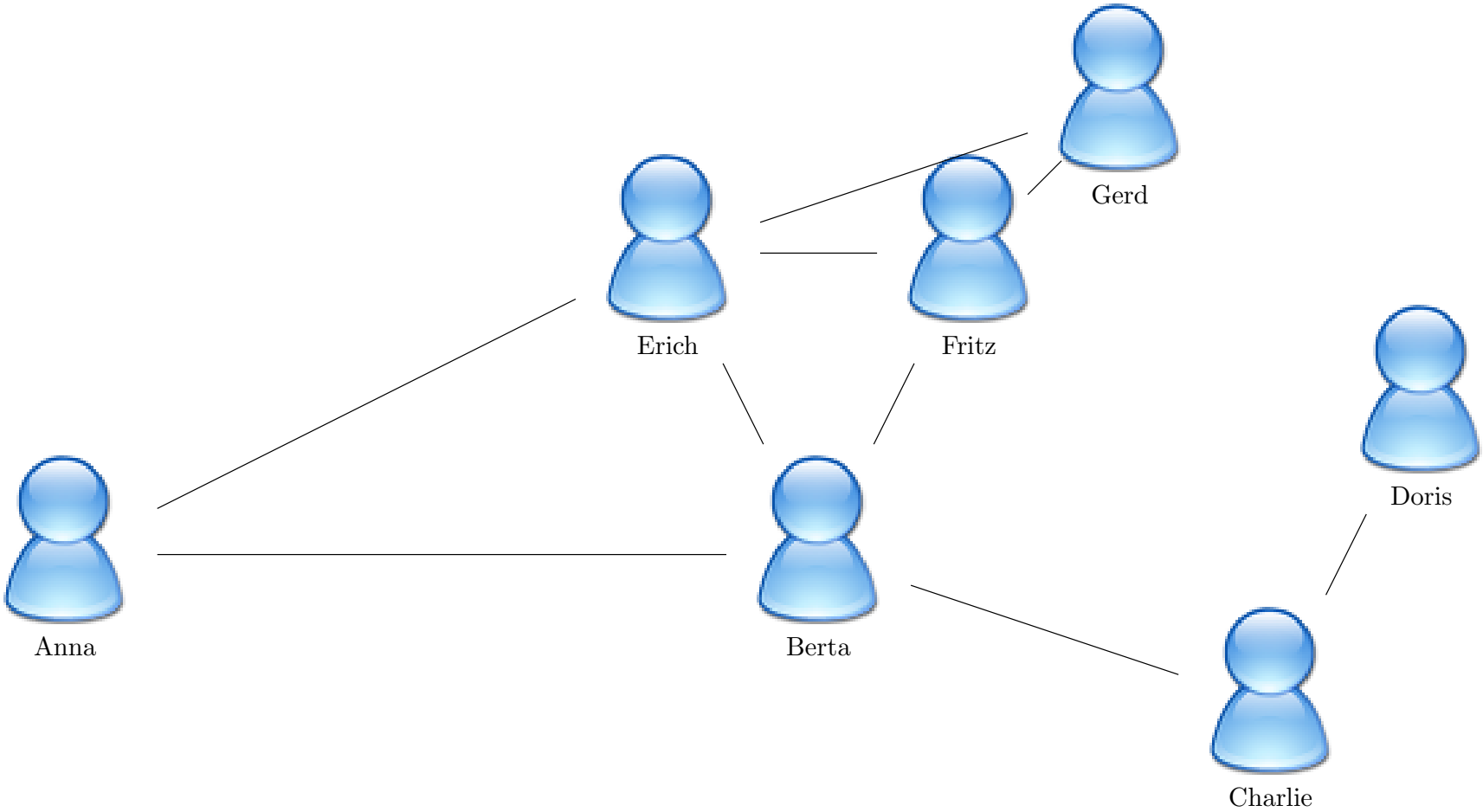
On Node Classification in Dynamic Content-based Networks

Martin Thoma | 28. Februar 2014

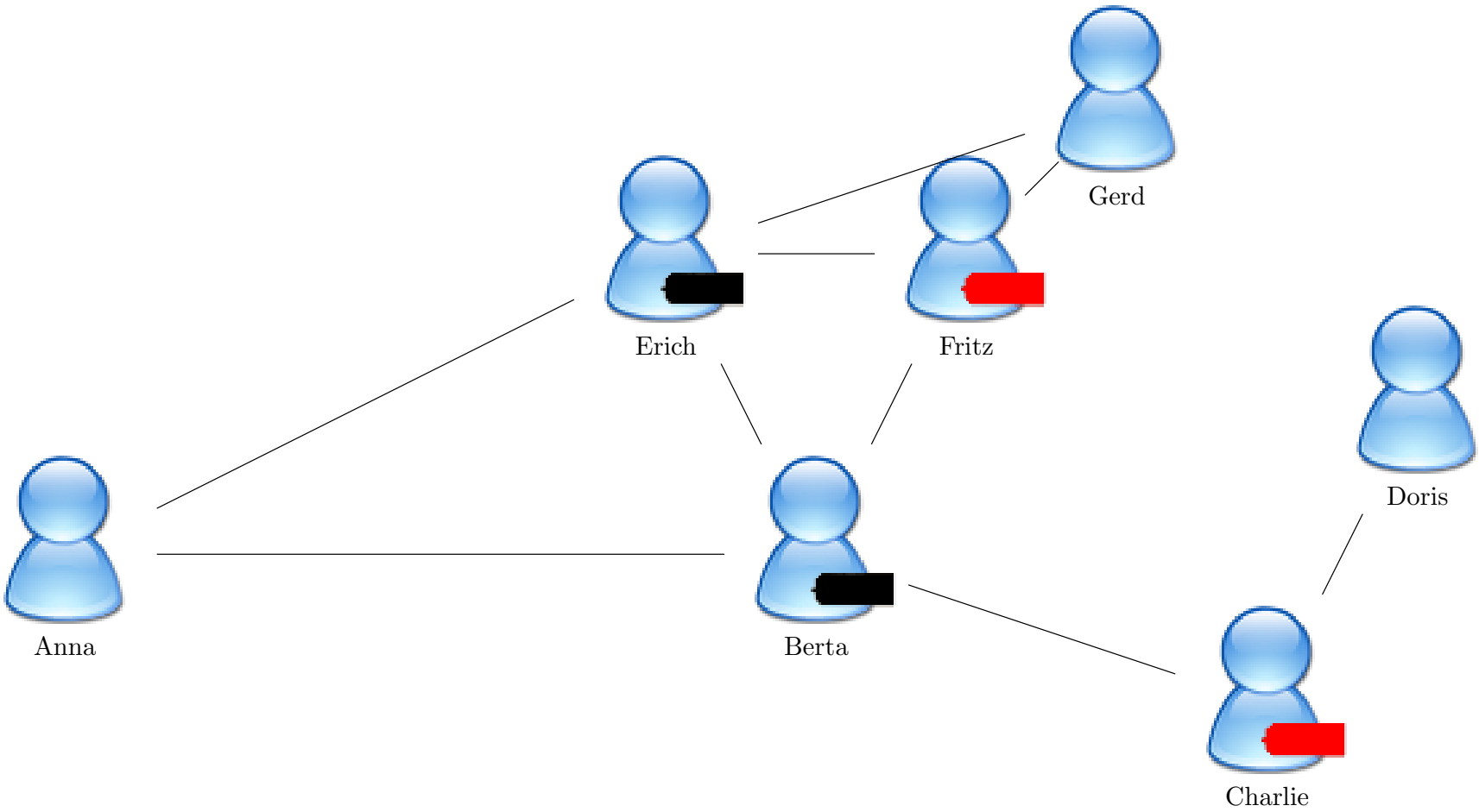
INSTITUT FÜR PROGRAMMSTRUKTUREN UND DATENORGANISATION



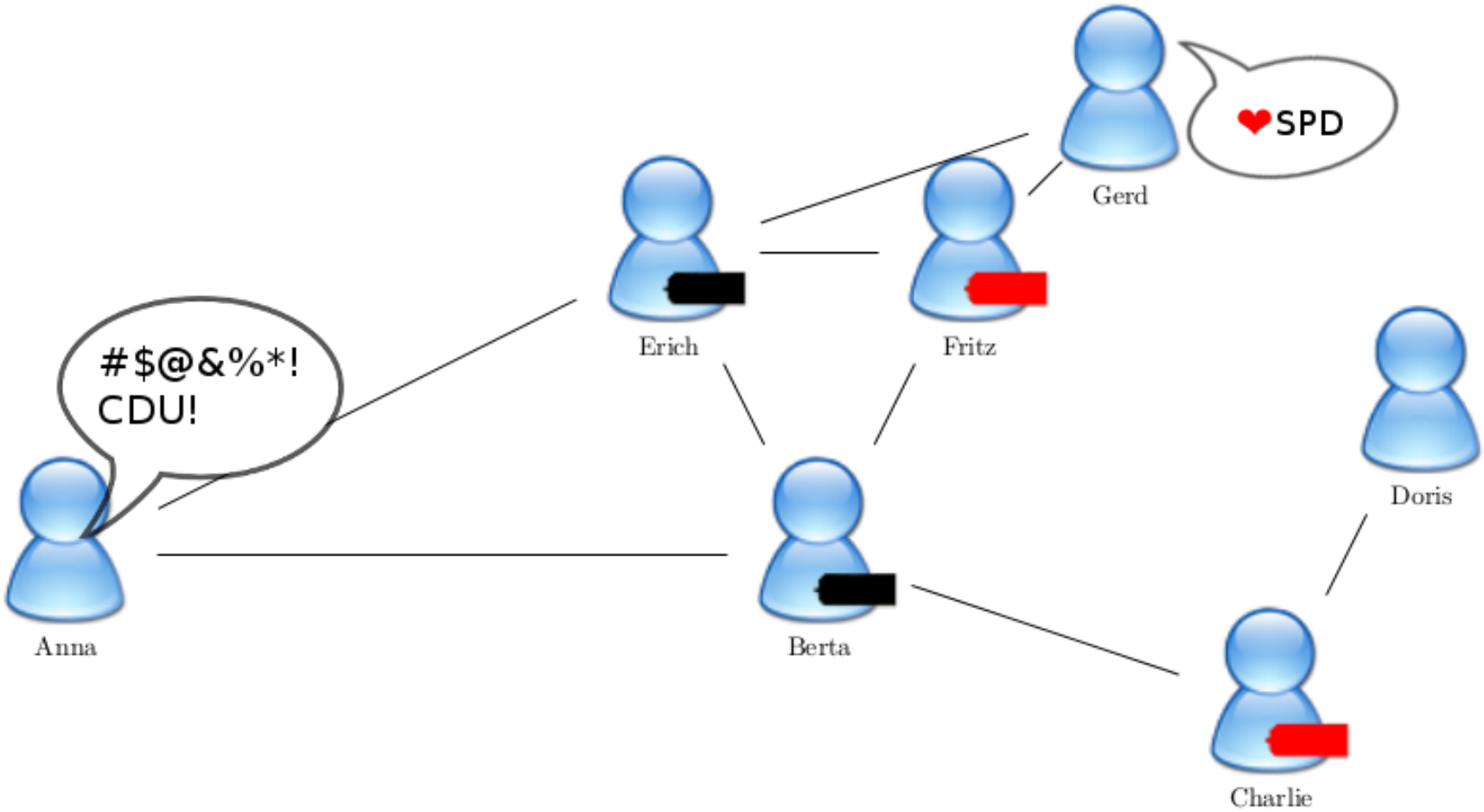
Social Network



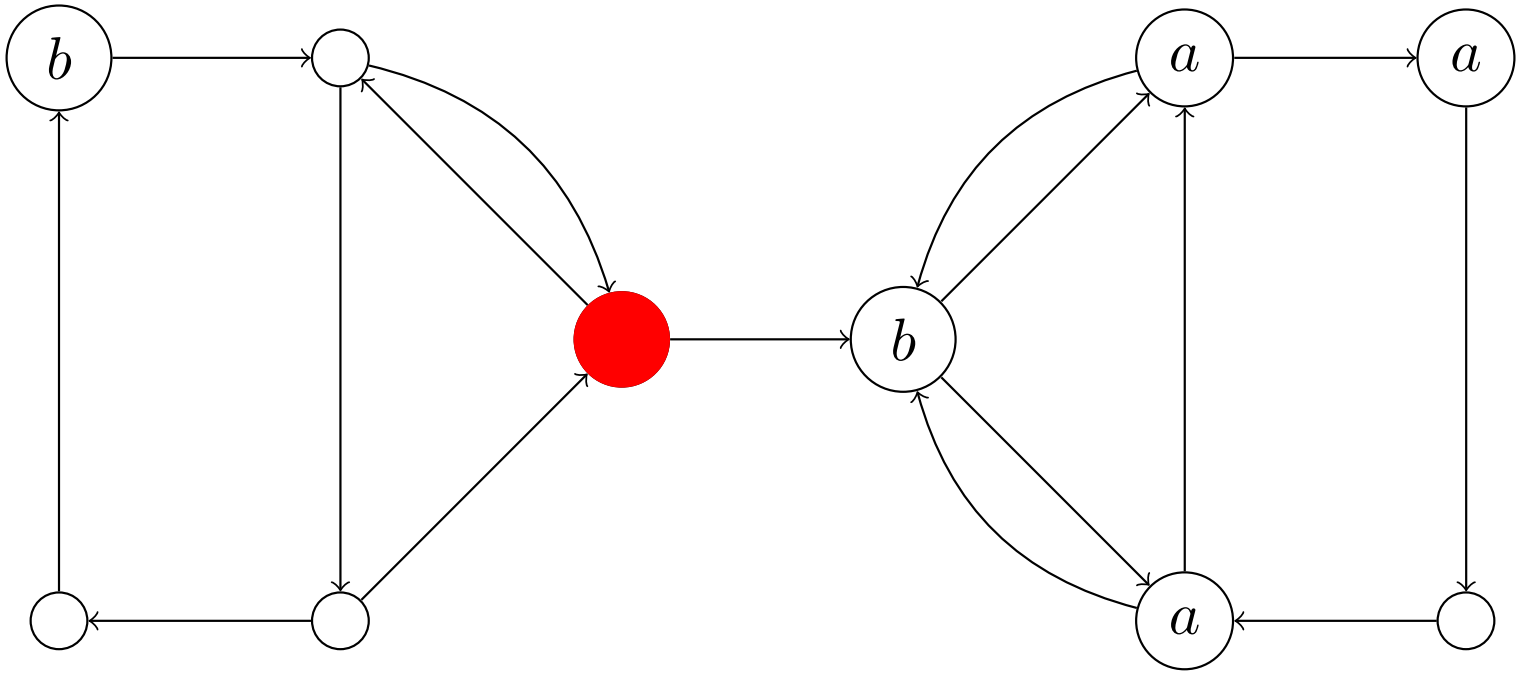
Partially labeled network



Partially labeled network with content



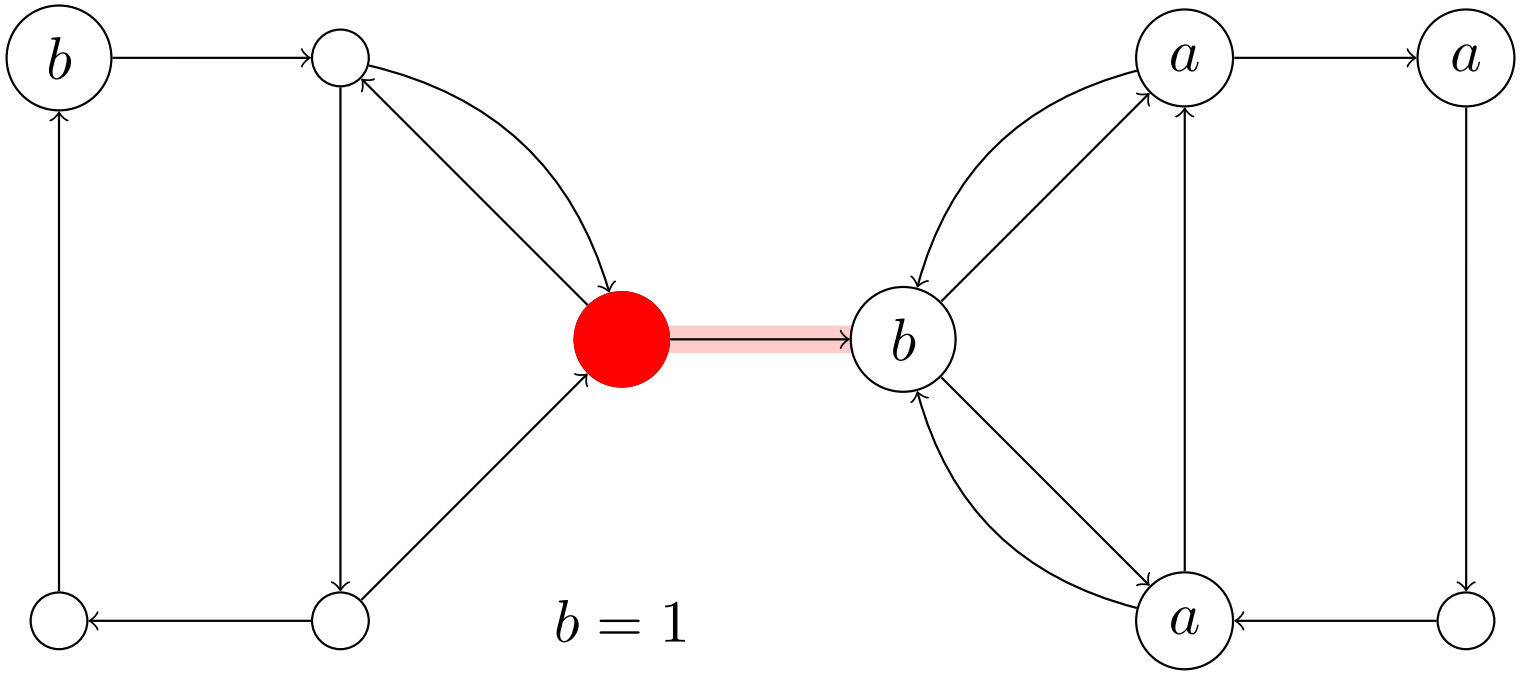
Random Walk



Klassifizieren des roten Knotens:

- Zählen von Knotenbeschriftungen in Random Walks
- 4 Random Walks
- 3 Sprünge pro Random Walk
- $4 \cdot a$, $2 \cdot b \Rightarrow$ Rot mit a klassifizieren

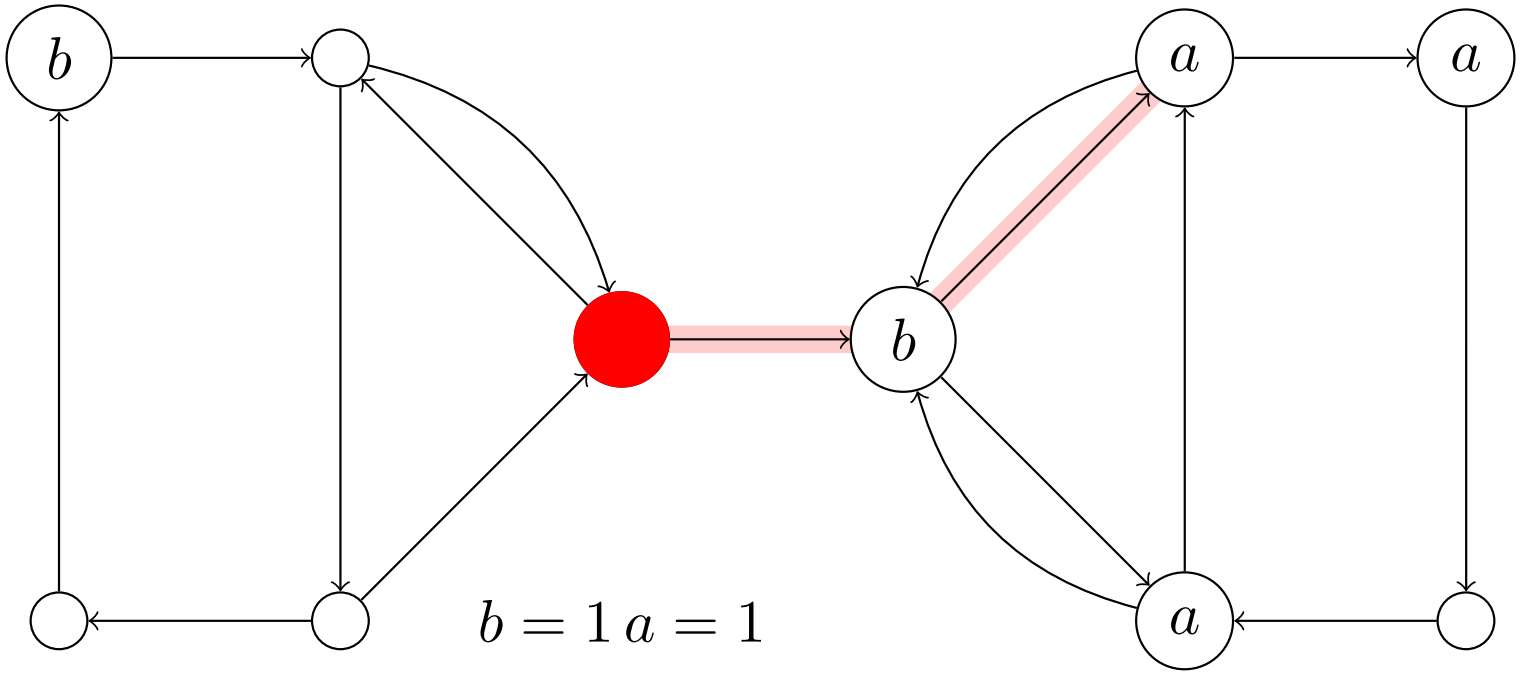
Random Walk



Klassifizieren des roten Knotens:

- Zählen von Knotenbeschriftungen in Random Walks
- 4 Random Walks
- 3 Sprünge pro Random Walk
- $4 \cdot a$, $2 \cdot b \Rightarrow$ Rot mit a klassifizieren

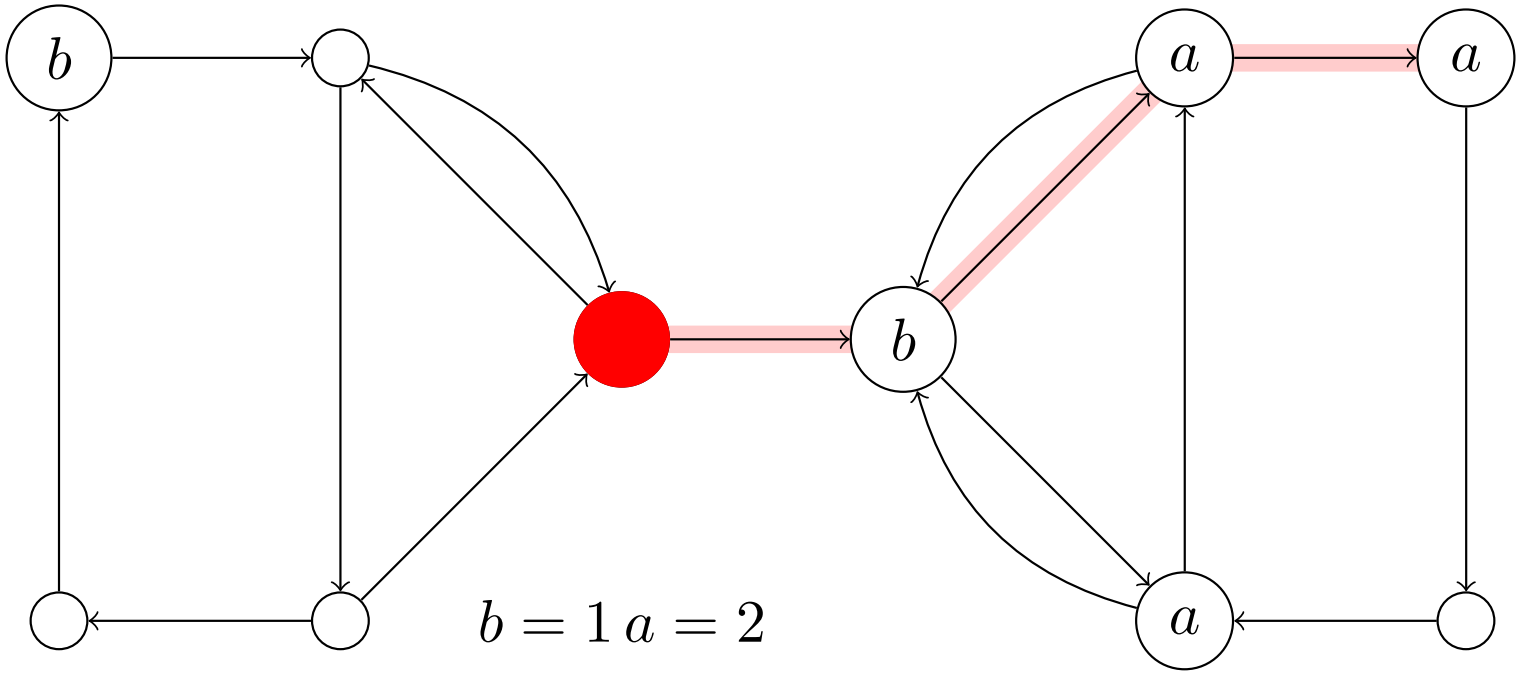
Random Walk



Klassifizieren des roten Knotens:

- Zählen von Knotenbeschriftungen in Random Walks
- 4 Random Walks
- 3 Sprünge pro Random Walk
- $4 \cdot a$, $2 \cdot b \Rightarrow$ Rot mit a klassifizieren

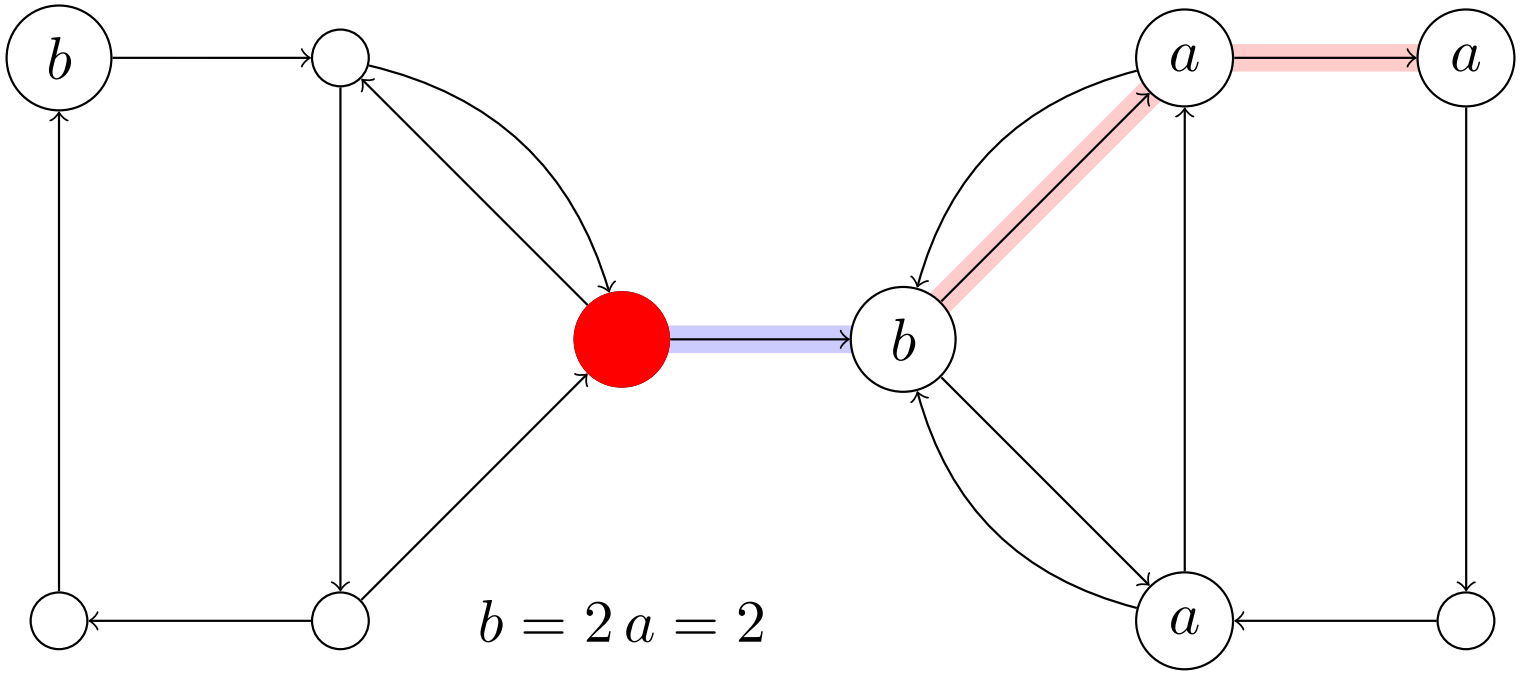
Random Walk



Klassifizieren des roten Knotens:

- Zählen von Knotenbeschriftungen in Random Walks
- 4 Random Walks
- 3 Sprünge pro Random Walk
- $4 \cdot a$, $2 \cdot b \Rightarrow$ Rot mit a klassifizieren

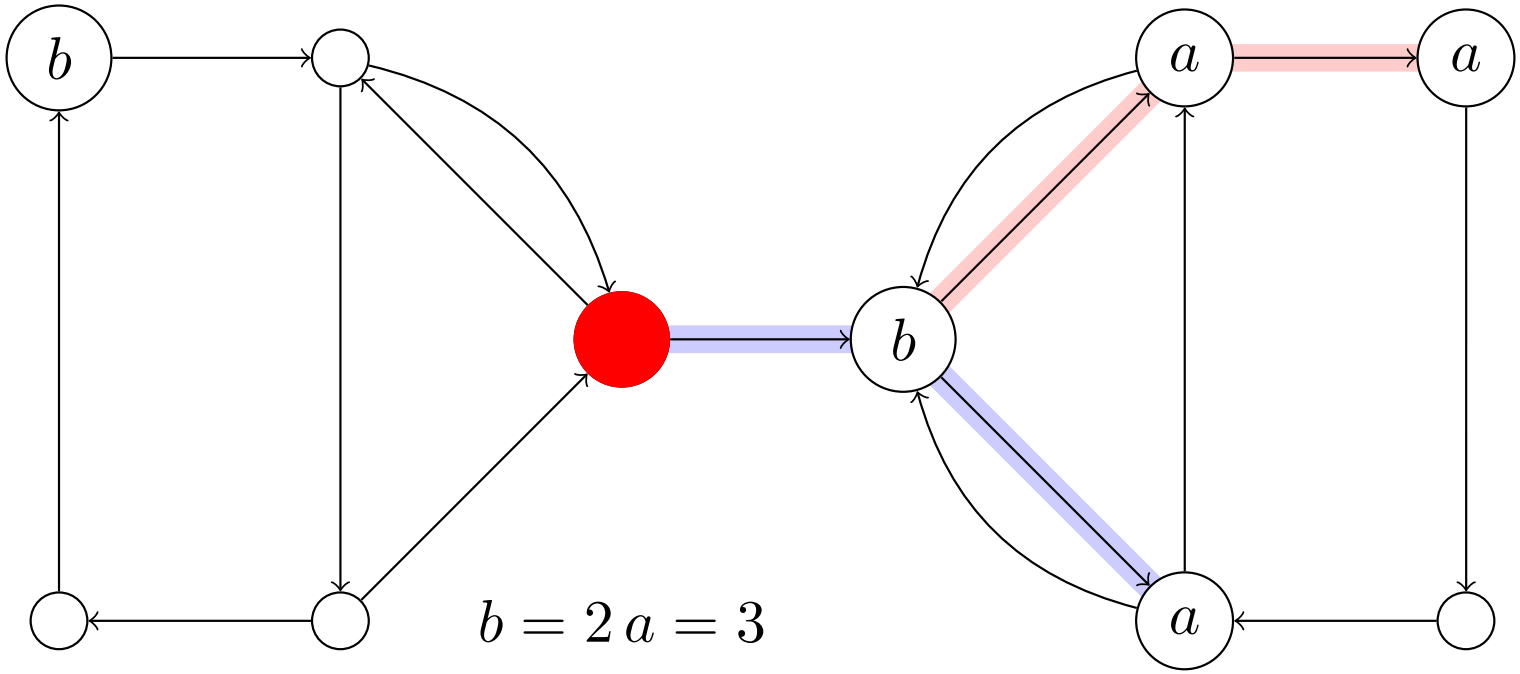
Random Walk



Klassifizieren des roten Knotens:

- Zählen von Knotenbeschriftungen in Random Walks
- 4 Random Walks
- 3 Sprünge pro Random Walk
- $4 \cdot a$, $2 \cdot b \Rightarrow$ Rot mit a klassifizieren

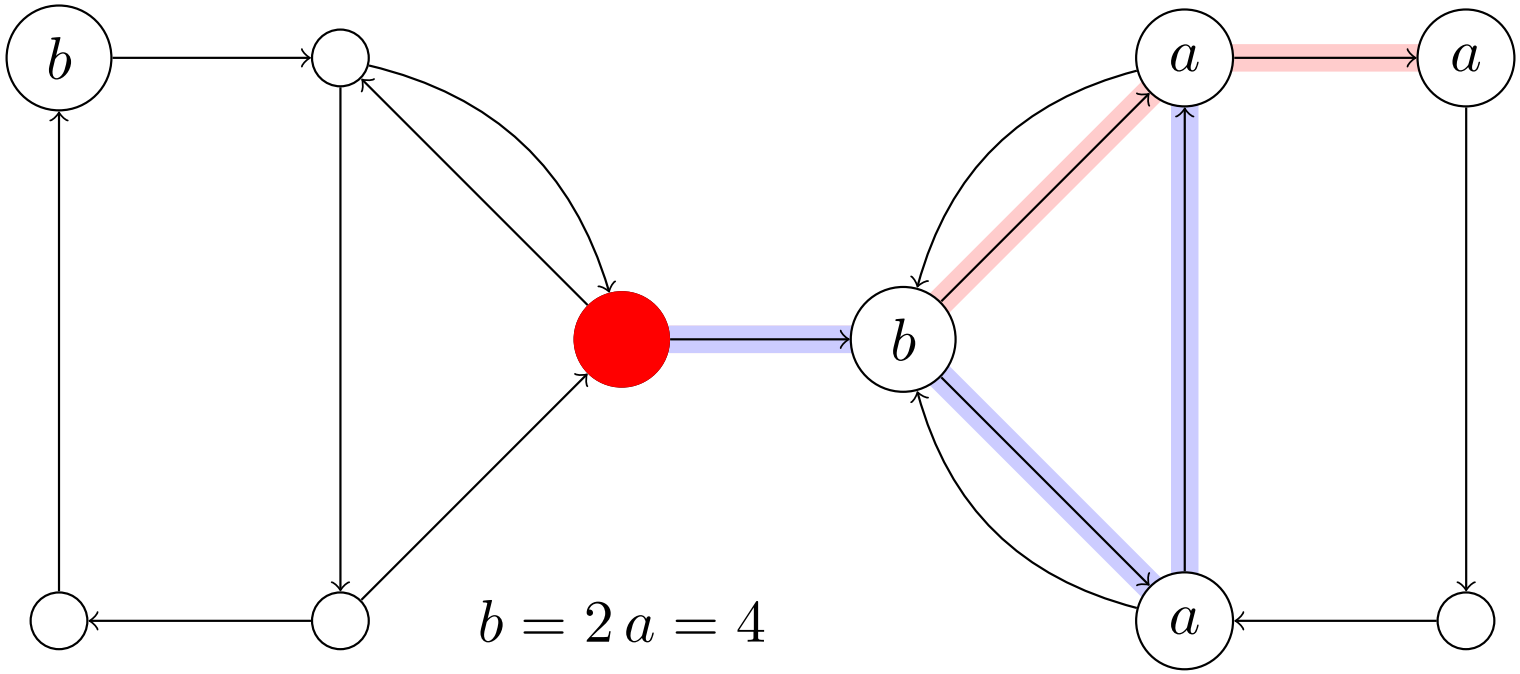
Random Walk



Klassifizieren des roten Knotens:

- Zählen von Knotenbeschriftungen in Random Walks
- 4 Random Walks
- 3 Sprünge pro Random Walk
- $4 \cdot a$, $2 \cdot b \Rightarrow$ Rot mit a klassifizieren

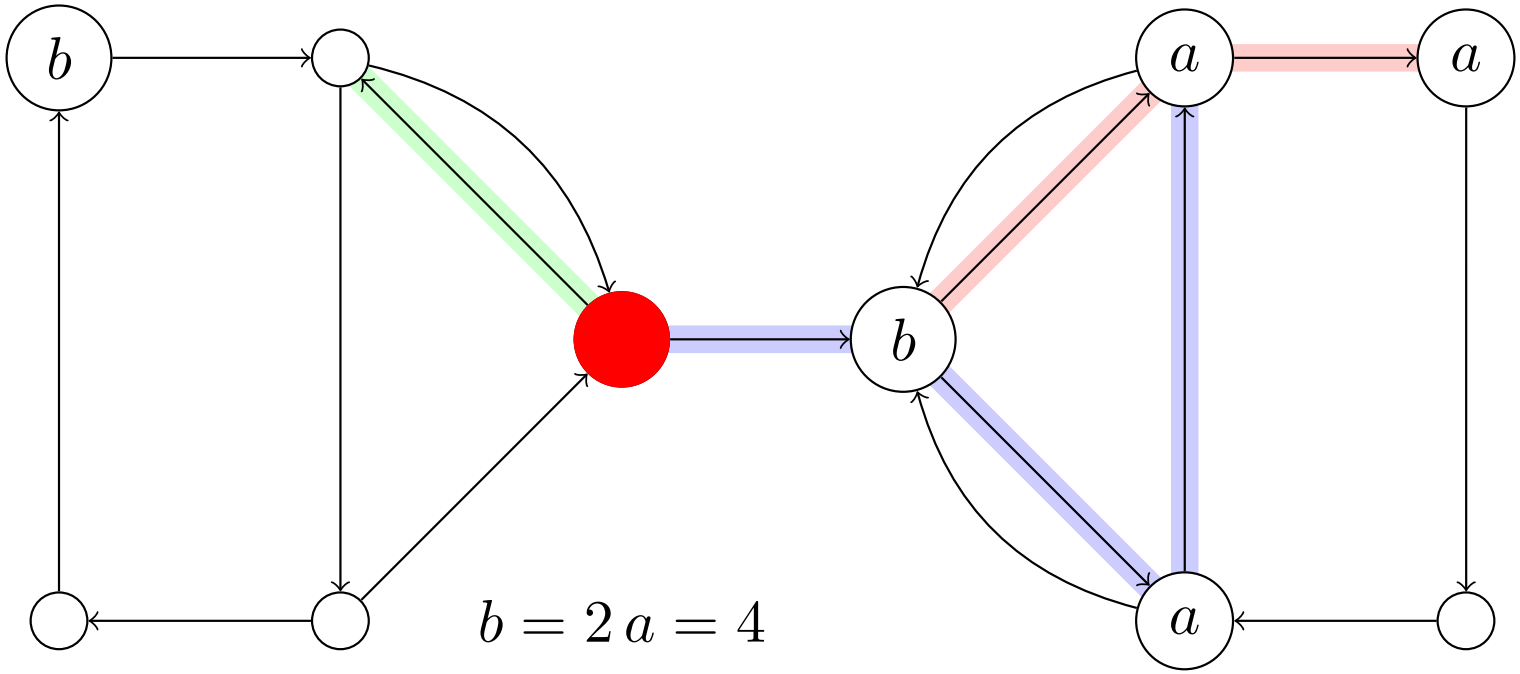
Random Walk



Klassifizieren des roten Knotens:

- Zählen von Knotenbeschriftungen in Random Walks
- 4 Random Walks
- 3 Sprünge pro Random Walk
- $4 \cdot a$, $2 \cdot b \Rightarrow$ Rot mit a klassifizieren

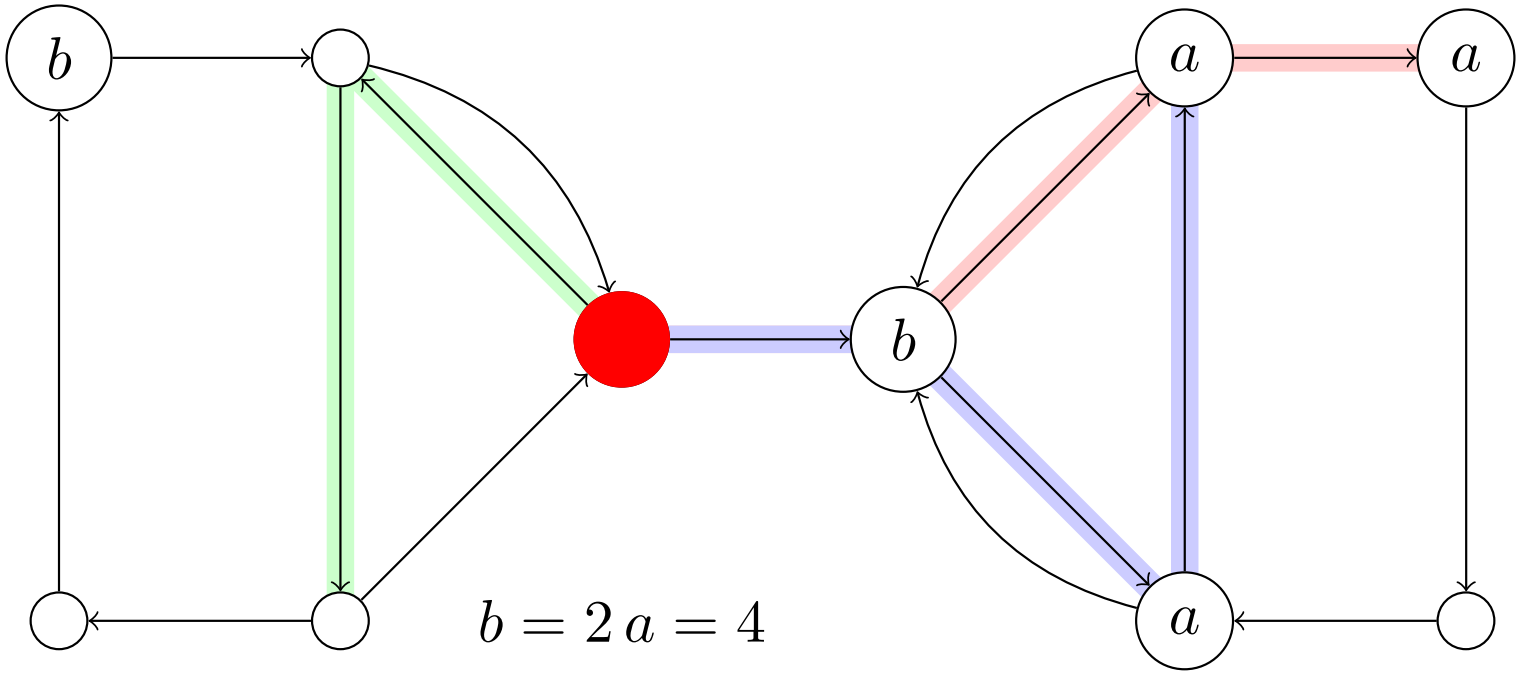
Random Walk



Klassifizieren des roten Knotens:

- Zählen von Knotenbeschriftungen in Random Walks
- 4 Random Walks
- 3 Sprünge pro Random Walk
- $4 \cdot a$, $2 \cdot b \Rightarrow$ Rot mit a klassifizieren

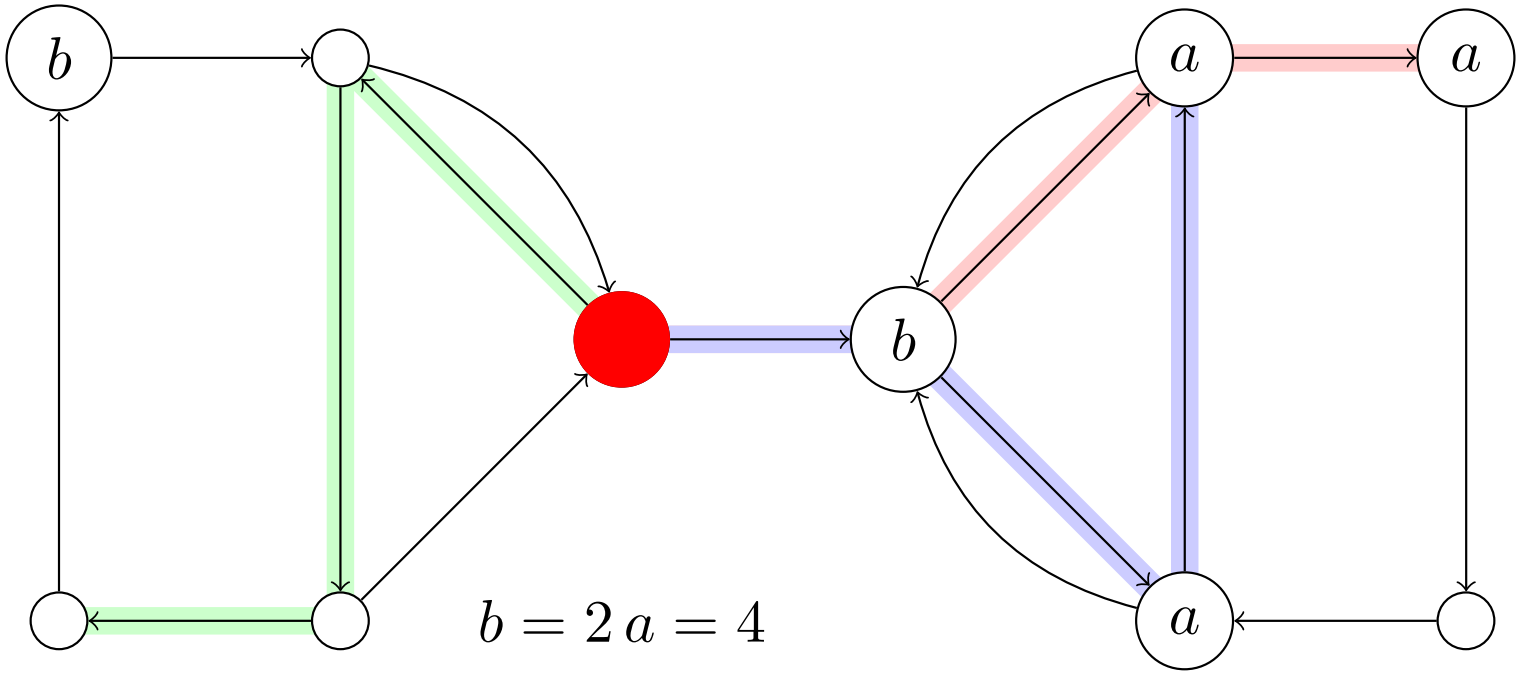
Random Walk



Klassifizieren des roten Knotens:

- Zählen von Knotenbeschriftungen in Random Walks
- 4 Random Walks
- 3 Sprünge pro Random Walk
- $4 \cdot a$, $2 \cdot b \Rightarrow$ Rot mit a klassifizieren

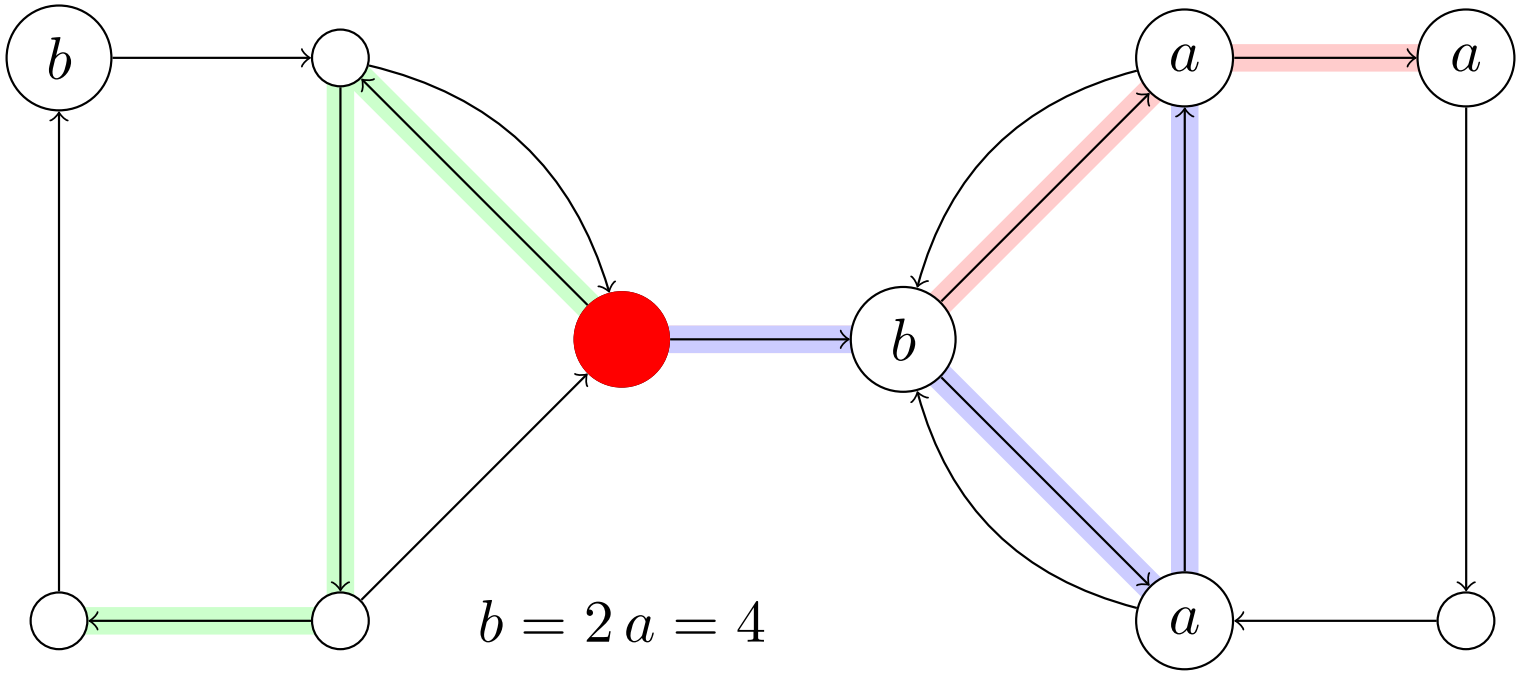
Random Walk



Klassifizieren des roten Knotens:

- Zählen von Knotenbeschriftungen in Random Walks
- 4 Random Walks
- 3 Sprünge pro Random Walk
- $4 \cdot a$, $2 \cdot b \Rightarrow$ Rot mit a klassifizieren

Random Walk



Klassifizieren des roten Knotens:

- Zählen von Knotenbeschriftungen in Random Walks
- 4 Random Walks
- 3 Sprünge pro Random Walk
- $4 \cdot a$, $2 \cdot b \Rightarrow$ Rot mit a klassifizieren

Wortknoten

- Neben Struktur können Texte genutzt werden
- Einschränkung: Effizienz!
- Wünschenswert: Wenig weiterer Programmieraufwand
- Idee: Graph erweitern
  - Texte als Wortmengen
  - Strukturknoten verweisen auf Wortknoten
  - vice versa

Wortknoten

- Neben Struktur können Texte genutzt werden
- Einschränkung: Effizienz!
- Wünschenswert: Wenig weiterer Programmieraufwand
- Idee: Graph erweitern
  - Texte als Wortmengen
  - Strukturknoten verweisen auf Wortknoten
  - vice versa

Wortknoten

- Neben Struktur können Texte genutzt werden
- Einschränkung: Effizienz!
- Wünschenswert: Wenig weiterer Programmieraufwand
- Idee: Graph erweitern
  - Texte als Wortmengen
  - Strukturknoten verweisen auf Wortknoten
  - vice versa

Wortknoten

- Neben Struktur können Texte genutzt werden
- Einschränkung: Effizienz!
- Wünschenswert: Wenig weiterer Programmieraufwand
- Idee: Graph erweitern
  - Texte als Wortmengen
  - Strukturknoten verweisen auf Wortknoten
  - vice versa

Wortknoten

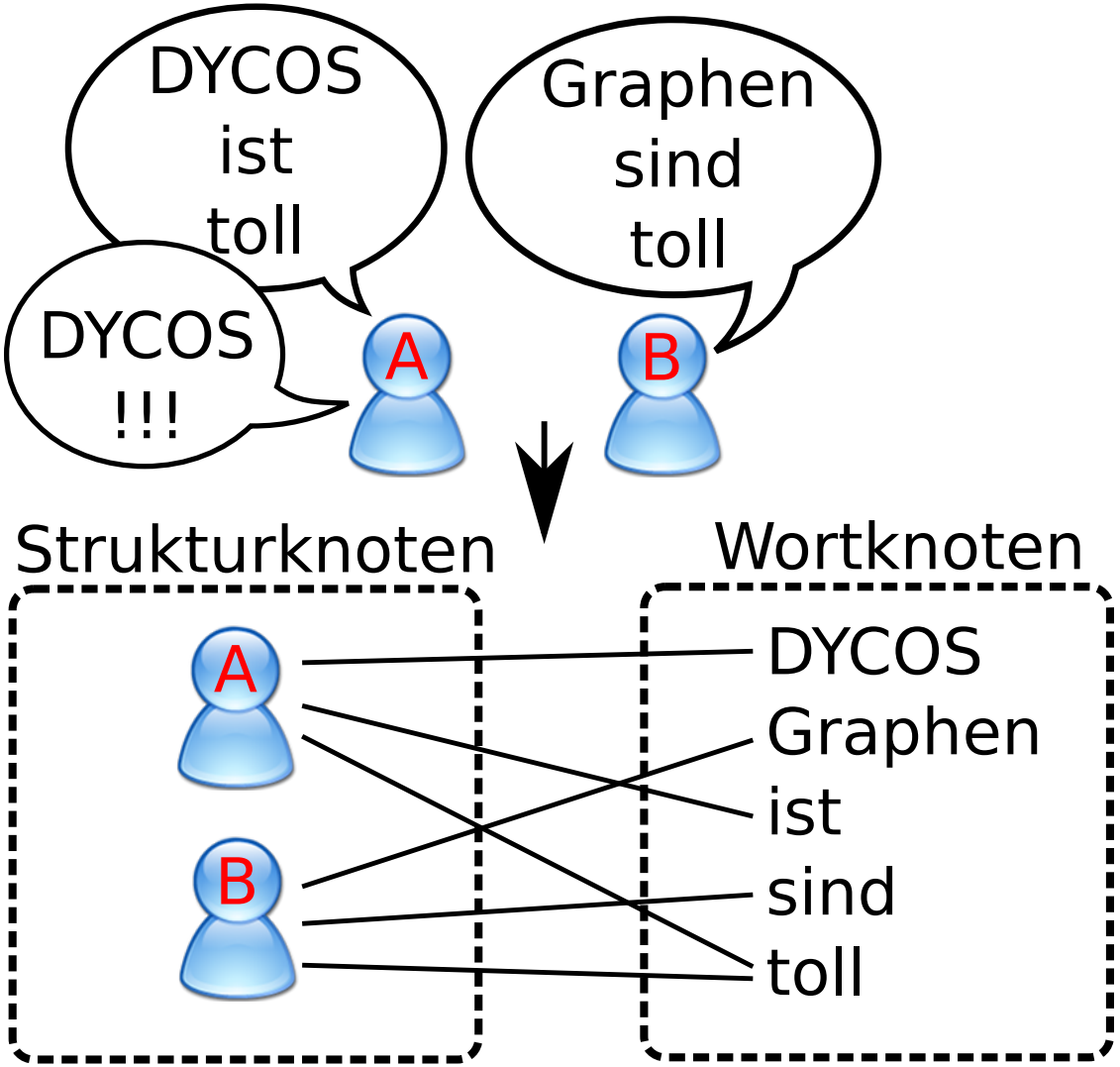
- Neben Struktur können Texte genutzt werden
- Einschränkung: Effizienz!
- Wünschenswert: Wenig weiterer Programmieraufwand
- Idee: Graph erweitern
  - Texte als Wortmengen
  - Strukturknoten verweisen auf Wortknoten
  - vice versa

Wortknoten

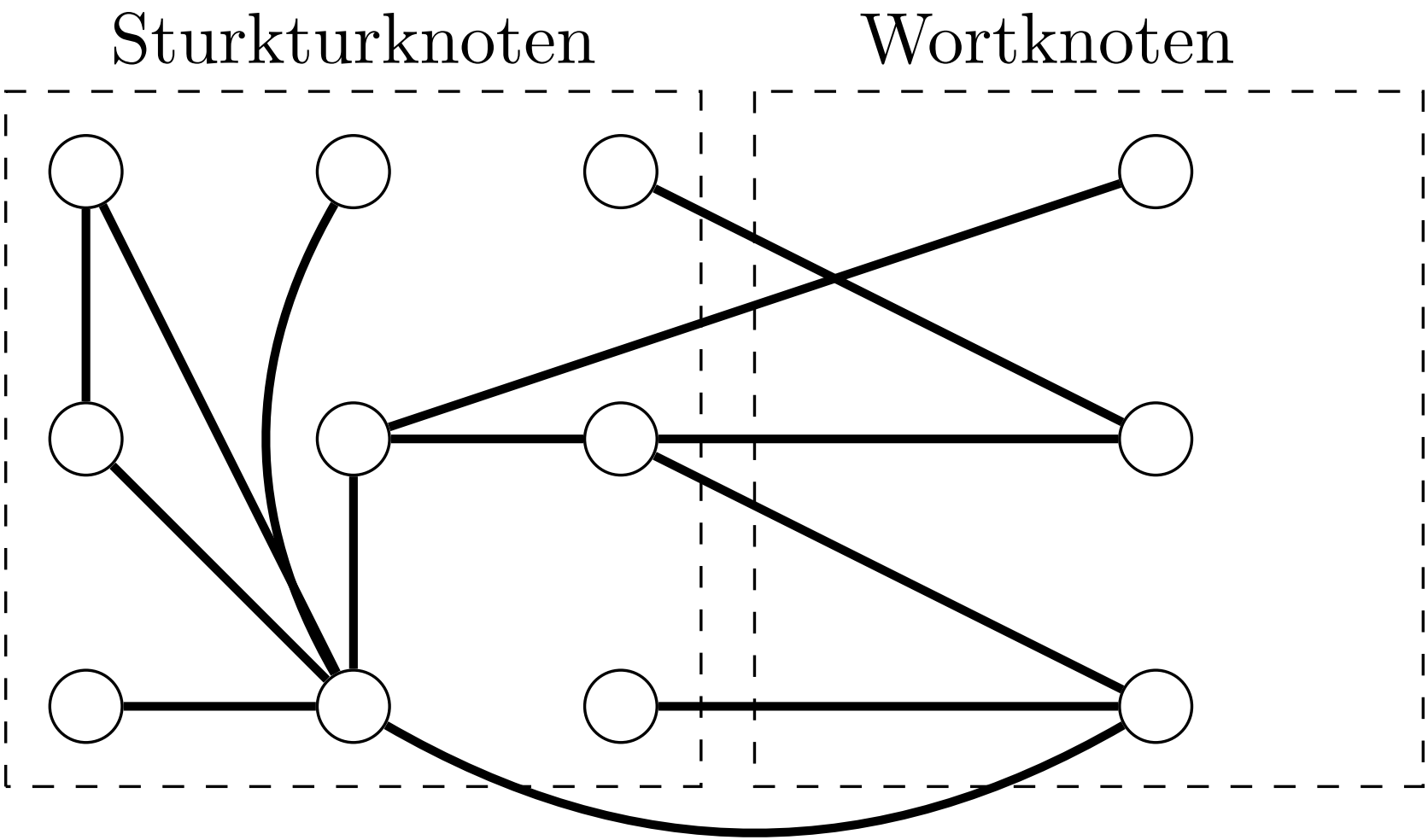
- Neben Struktur können Texte genutzt werden
- Einschränkung: Effizienz!
- Wünschenswert: Wenig weiterer Programmieraufwand
- Idee: Graph erweitern
  - Texte als Wortmengen
  - Strukturknoten verweisen auf Wortknoten
  - vice versa

Wortknoten

- Neben Struktur können Texte genutzt werden
- Einschränkung: Effizienz!
- Wünschenswert: Wenig weiterer Programmieraufwand
- Idee: Graph erweitern
  - Texte als Wortmengen
  - Strukturknoten verweisen auf Wortknoten
  - vice versa


DYCOS ist toll
Graphen sind toll
DYCOS !!!
A
B
Strukturknoten
Wortknoten
A
B
DYCOS
Graphen
ist
sind
toll

Erweiterter, semi-bipartiter Graph



Vokabular

- Viele Texte $\Rightarrow$ Komplette Textanalyse nicht möglich
- Füllwörter: und, oder, im, in, ...

$\Rightarrow$ Beschränkung des Vokabulars sinnvoll

Idee:

- Zufällige Beispielmenge von Texten für Vokabularbildung betrachten
- Gini-Koeffizient nutzen

Vokabular

- Viele Texte $\Rightarrow$ Komplette Textanalyse nicht möglich
- Füllwörter: und, oder, im, in, ...

$\Rightarrow$ Beschränkung des Vokabulars sinnvoll

Idee:

- Zufällige Beispielmenge von Texten für Vokabularbildung betrachten
- Gini-Koeffizient nutzen

Vokabular

- Viele Texte $\Rightarrow$ Komplette Textanalyse nicht möglich
- Füllwörter: und, oder, im, in, ...

$\Rightarrow$ Beschränkung des Vokabulars sinnvoll

Idee:

- Zufällige Beispielmenge von Texten für Vokabularbildung betrachten
- Gini-Koeffizient nutzen

Vokabular

- Viele Texte $\Rightarrow$ Komplette Textanalyse nicht möglich
- Füllwörter: und, oder, im, in, ...

$\Rightarrow$ Beschränkung des Vokabulars sinnvoll

Idee:

- Zufällige Beispielmenge von Texten für Vokabularbildung betrachten
- Gini-Koeffizient nutzen

Vokabular

- Viele Texte $\Rightarrow$ Komplette Textanalyse nicht möglich
- Füllwörter: und, oder, im, in, ...

$\Rightarrow$ Beschränkung des Vokabulars sinnvoll

Idee:

- Zufällige Beispielmenge von Texten für Vokabularbildung betrachten
- Gini-Koeffizient nutzen

Vokabular

- Viele Texte $\Rightarrow$ Komplette Textanalyse nicht möglich
- Füllwörter: und, oder, im, in, ...

$\Rightarrow$ Beschränkung des Vokabulars sinnvoll

Idee:

- Zufällige Beispielmenge von Texten für Vokabularbildung betrachten
- Gini-Koeffizient nutzen

Gini-Koeffizient

- statistisches Maß für Ungleichverteilung
- $g = \sum_i p_i^2$ mit p_i als relative Häufigkeit
- $g \in (0, 1]$
- g nahe bei $1 \Rightarrow$ Wort ist stark ungleich verteilt

$\Rightarrow$ Nehme Top-m Wörter mit höchstem Gini-Koeffizient

Gini-Koeffizient

- statistisches Maß für Ungleichverteilung
- $g = \sum_i p_i^2$ mit p_i als relative Häufigkeit
- $g \in (0, 1]$
- g nahe bei $1 \Rightarrow$ Wort ist stark ungleich verteilt

$\Rightarrow$ Nehme Top-m Wörter mit höchstem Gini-Koeffizient

Gini-Koeffizient

- statistisches Maß für Ungleichverteilung
- $g = \sum_i p_i^2$ mit p_i als relative Häufigkeit
- $g \in (0, 1]$
- g nahe bei $1 \Rightarrow$ Wort ist stark ungleich verteilt

$\Rightarrow$ Nehme Top-m Wörter mit höchstem Gini-Koeffizient

Gini-Koeffizient

- statistisches Maß für Ungleichverteilung
- $g = \sum_i p_i^2$ mit p_i als relative Häufigkeit
- $g \in (0, 1]$
- g nahe bei $1 \Rightarrow$ Wort ist stark ungleich verteilt

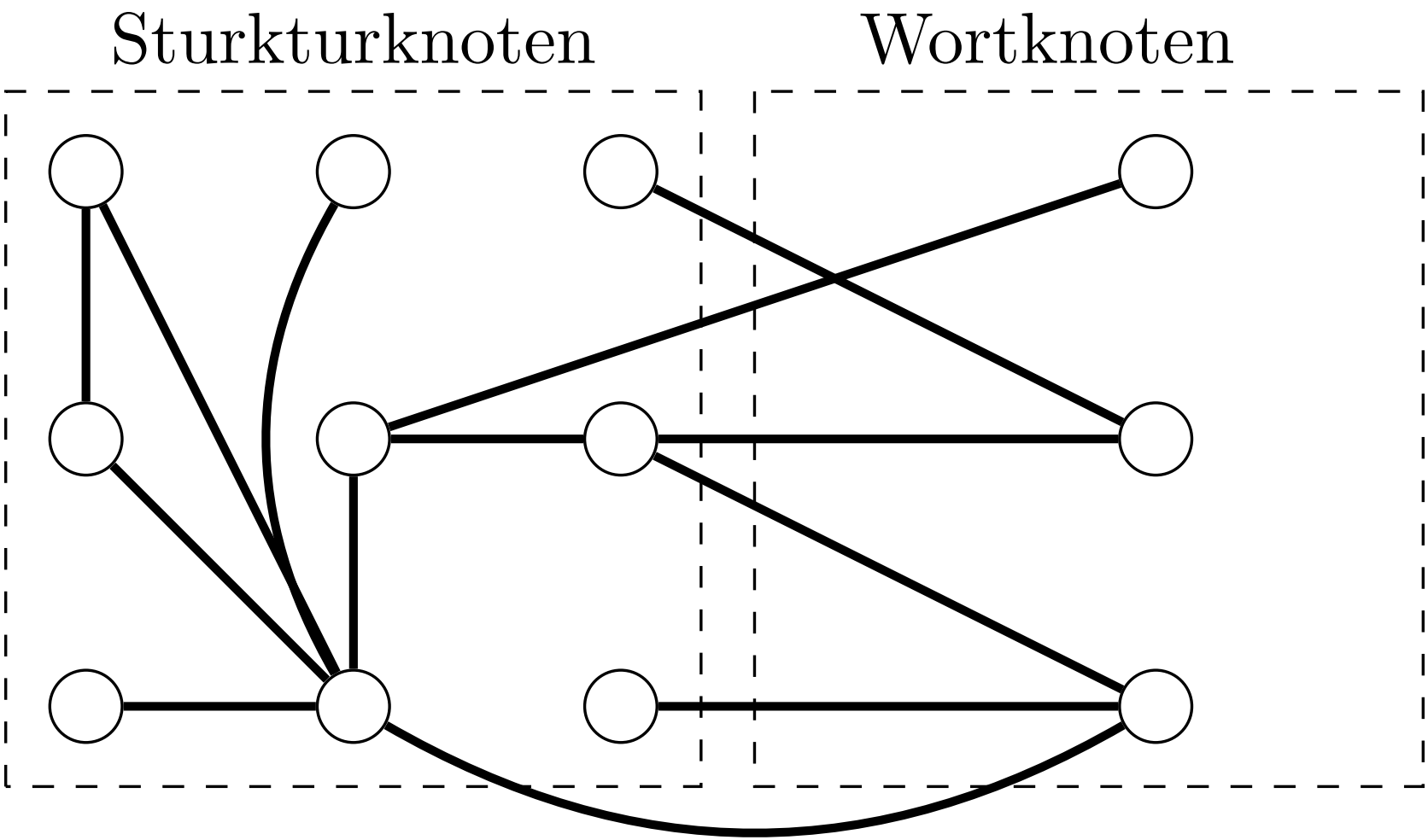
$\Rightarrow$ Nehme Top-m Wörter mit höchstem Gini-Koeffizient

Gini-Koeffizient

- statistisches Maß für Ungleichverteilung
- $g = \sum_i p_i^2$ mit p_i als relative Häufigkeit
- $g \in (0, 1]$
- g nahe bei $1 \Rightarrow$ Wort ist stark ungleich verteilt

$\Rightarrow$ Nehme Top-m Wörter mit höchstem Gini-Koeffizient

Sprungtypen



Inhaltlicher Mehrfachsprung

- **Strukturspung**: von Strukturknoten v zu Strukturknoten v'
- **Inhaltlicher Mehrfachsprung**: von Strukturknoten v über Wortknoten zu Strukturknoten v'

Inhaltlicher Mehrfachsprung

- **Strukturspung**: von Strukturknoten v zu Strukturknoten v'
- **Inhaltlicher Mehrfachsprung**: von Strukturknoten v über Wortknoten zu Strukturknoten v'

Danke!

Gibt es Fragen?

Bildquellen

- Crystal_Clear_app_personal.png von Wikipedia Commons

Literatur

- Charu C. Aggarwal, Nan Li: *On Node Classification in Dynamic Content-based Networks*
- Smriti Bhagat, Graham Cormode und S. Muthukrishnan. *Node Classification in Social Networks*
- M. F. Porter. Readings in Information Retrieval. Kapitel *An Algorithm for Suffix Stripping*
- Jeffrey S. Vitter. *Random Sampling with a Reservoir*.

Folien, LaTeXund Material

Der Foliensatz und die LaTeXund TikZ-Quellen sind unter
github.com/MartinThoma/LaTeX-examples/tree/master/presentations/Datamining-Proseminar
Kurz-URL: tinyurl.com/Info-Proseminar