Algorithmen und Datenstrukturen

Zeichenerkennung, Subgraph-Isomorphie, Pruning von Suchräumen

Prof. Dr. Ralf Möller

Universität zu Lübeck

Institut für Informationssysteme

Felix Kuhr (Übungen)

sowie viele Tutoren

Danksagung

Nach einem Vortrag von

Optical Character Recognition:
Using the Ullmann Algorithm for Graphical Matching

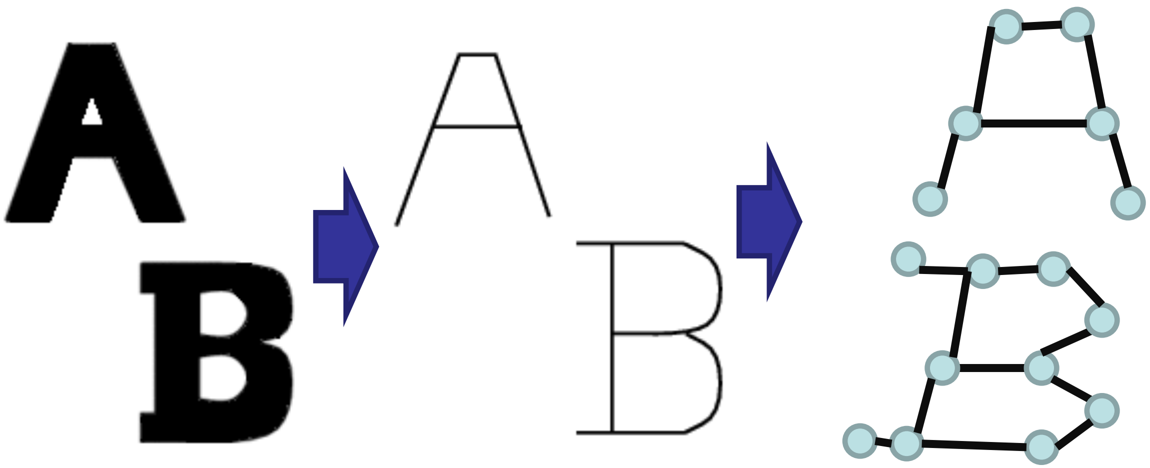
von

Iddo Aviram

OCR – Zeichenerkennung durch Graphabgleich

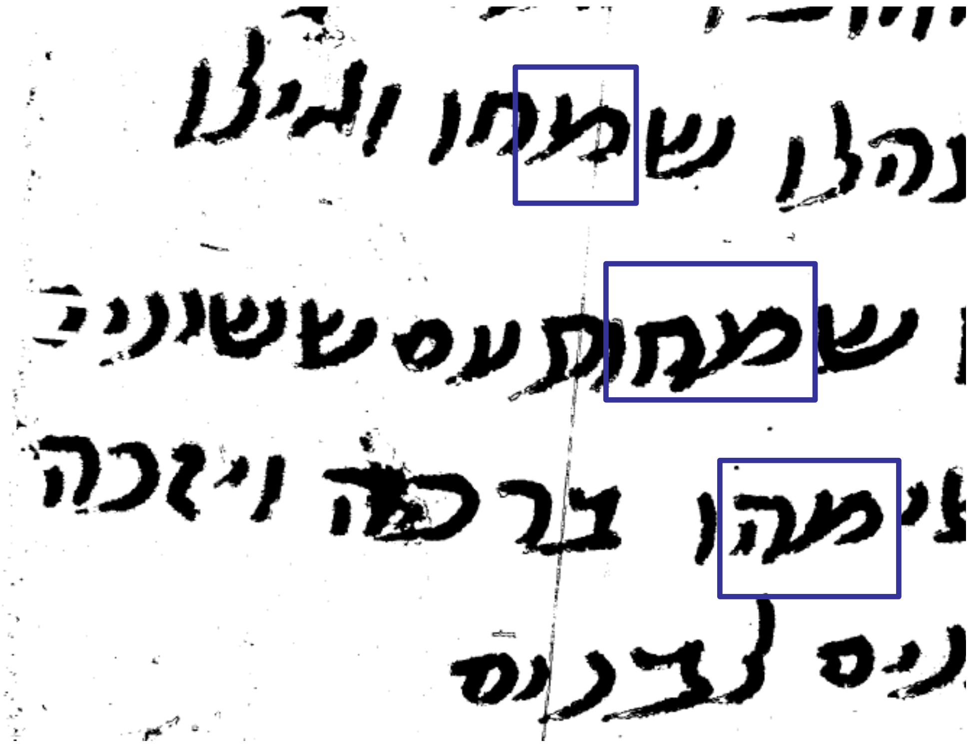
Bestimmung einer Datenstruktur:

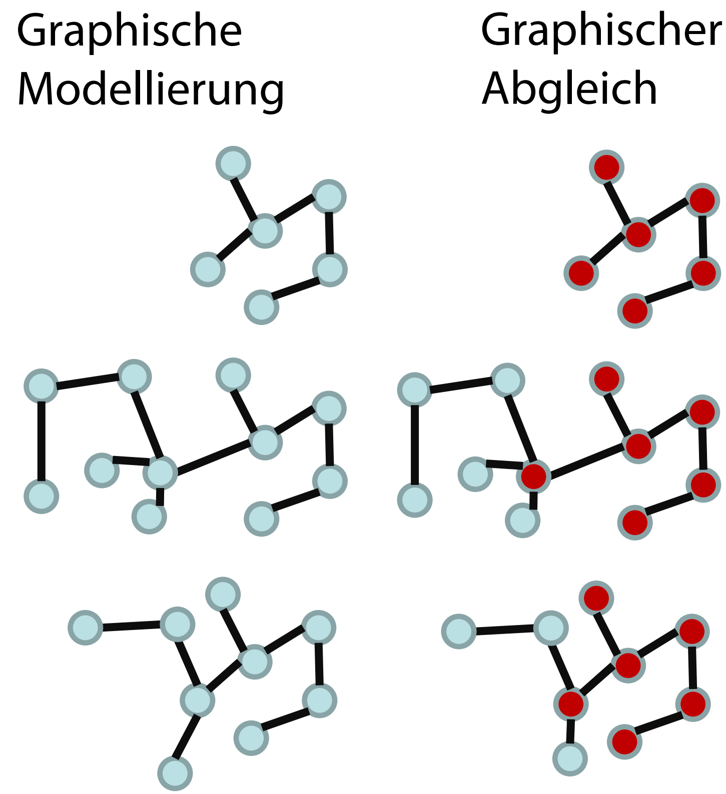
Segmentation ⟶ Verdünnung ⟶ Graphrepräsentation



OCR durch Graphabgleich (Matching)

Durch Subgraphabgleich Kandidaten finden

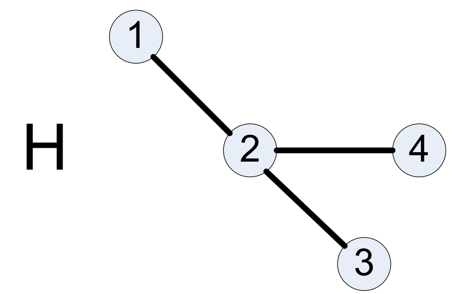


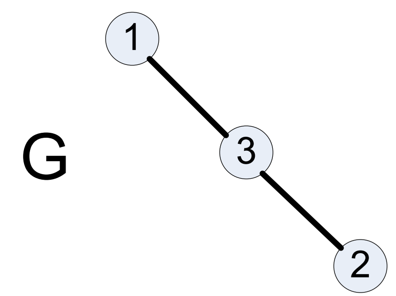


Subgraph-Isomorphie

Subgraph-Isomorphismus-Problem

- Gegeben zwei Graphen H and G. Bestimme, ob H einen Subgraphen hat, der isomorph zu G ist, also bis auf Knotenumbenennung die Gestalt von G hat
- Zur Lösung müssen wir Korrespondenzen finden
- Beispiel: (Es gibt weitere Lösungen)
  - $\mathbf{1}_G\text{-}\mathbf{1}_H$
    $\mathbf{2}_G\text{-}\mathbf{3}_H$
    $\mathbf{3}_G\text{-}\mathbf{2}_H$
  - Antwort: Ja
- Sollen wir alle möglichen Korrespondenzen generieren und testen?

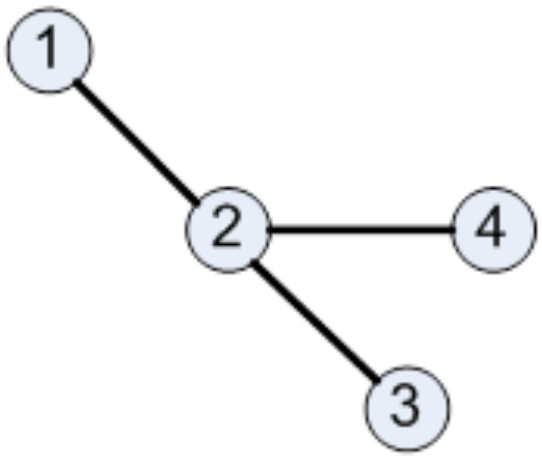




Ullmanns Algorithmus

- Verwendung von algebraischen Formulierungen des Subgraph-Isomorphie-Problems
- Adjazenzmatrix A_H eines Graphen H:

H



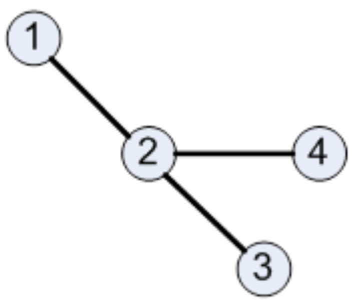
$$A_H = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

J. R. Ullmann, An Algorithm for Subgraph Isomorphism,
Journal of the ACM, **1976**

Aufbau des Suchraums nach Korrespondenzen

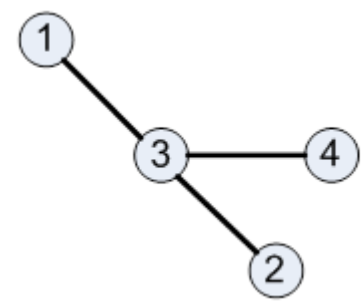
- Verwendung der sog. Permutationsmatrix
- Über Permutationsmatrix kann isomorphe Korrespondenz ausgedrückt werden

H



$$A_H = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

H_2



Isomorphe Korrespondenz

$$F = \begin{matrix} \mathbf{1}_{H_2} - \mathbf{1}_H \\ \mathbf{2}_{H_2} - \mathbf{3}_H \\ \mathbf{3}_{H_2} - \mathbf{2}_H \\ \mathbf{4}_{H_2} - \mathbf{4}_H \end{matrix}$$

Permutationsmatrix

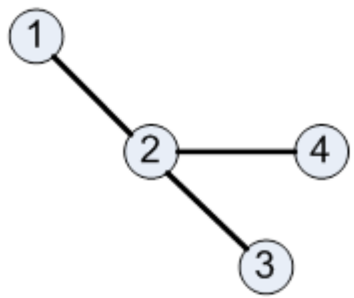
$$P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$F \sim P$

Isomorphiekriterium

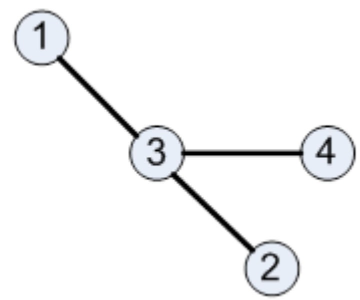
Zwei Graphen H und H_2 sind isomorph mit Korrespondenz F genau dann, wenn A_{H} ähnlich zu $\mathrm{A}_{\mathrm{H}_2}$ ist, also eine Permutationsmatrix P~F existiert.

H



$$A_H = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

H_2



Isomorphe Korrespondenz ~ Permutationsmatrix

F= $\mathbf{1}_{H_2}$- $\mathbf{1}_H$
$\mathbf{2}_{H_2}$- $\mathbf{3}_H$
$\mathbf{3}_{H_2}$- $\mathbf{2}_H$
$\mathbf{4}_{H_2}$- $\mathbf{4}_H$

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

F~P

Isomorphiekriterium:

$$A_{H_2} = P A_H P^{\,T}$$

gdw H_2 ist isomorph zu H mit Korrespondenz F~P

Skeptisch?

$$P=\begin{bmatrix}1&0&0&0\\0&0&1&0\\0&1&0&0\\0&0&0&1\end{bmatrix}\qquad A_H=\begin{bmatrix}0&1&0&0\\1&0&1&1\\0&1&0&0\\0&1&0&0\end{bmatrix}$$

P "moves" the row and columns of A_H using the mapping $\begin{pmatrix}1&2&3&4\\1&3&2&4\end{pmatrix}$

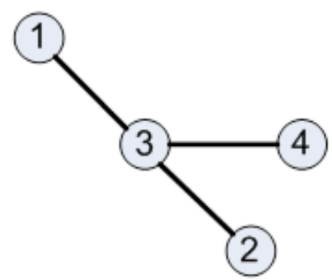
1: Moves only the rows

$$PA_H=\begin{bmatrix}0&1&0&0\\0&1&0&0\\1&0&1&1\\0&1&0&0\end{bmatrix}$$

2: Moves only the columns

$$PA_HP^T=\begin{bmatrix}0&0&1&0\\0&0&1&0\\1&1&0&1\\0&0&1&0\end{bmatrix}=A_{H_2}$$

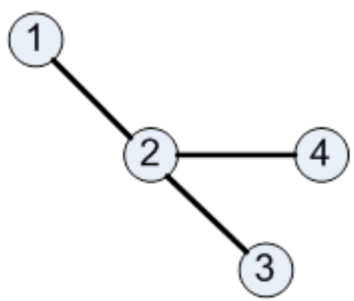
H_2



Ullmanns Algorithmus

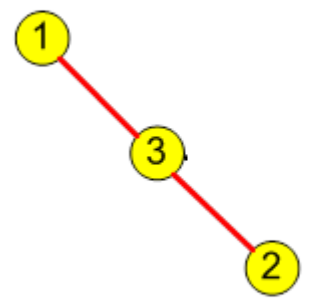
In gleicher Weise können wir ein algebraisches Kriterium für Subgraph-Isomorphie bestimmen

H



$$A_H = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

G



Isomorphe Korrespondenz ~ Permutationsmatrix

F= **1**$_G$-**1**$_H$
2$_G$-**3**$_H$
3$_G$-**2**$_H$
4$_G$-φ

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

Eine 1 pro Zeile

Ullmanns Algorithmus

- Gegeben: Graphen G und H
- Gesucht: Permutationsmatrix P mit Dimensionen $|V_G|$x $|V_H|$, so dass das Subgraph-Isomorphie-Kriterium erfüllt

- Suchraum über alle möglichen Permutationsmatrizen P mit jeweiliger Prüfung des Kriteriums aufspannen?

- Das geht besser!

Ullmanns Algorithmus

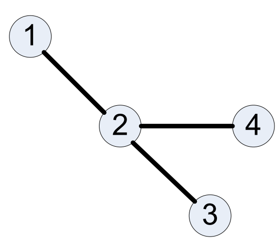
- Konstruiere Matrix $M^{(0)}$ der Dimension der P-Matrizen:

$$m_{i,j}^{(0)} = \begin{cases} 1 & \text{if } \deg(V_{Hj}) \geq \deg(V_{Gi}) \\ 0 & \text{otherwise} \end{cases}, m_{i,j} \in \{0,1\}$$

- Generiere aus $M^{(0)}$ alle P durch Wahl einer 1 pro Zeile
- Subgraph-Isomorphismus gefunden, wenn

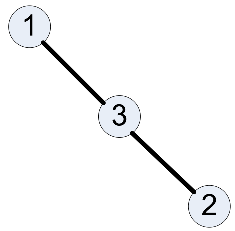
$$A_G = P A_H P^T$$

H



$$A_H = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \Rightarrow M^0 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

G



$$A_G = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

Ullmanns Algorithmus (mit einfachem Pruning)

$$\begin{bmatrix}1&1&1&1\\1&1&1&1\\0&1&0&0\end{bmatrix}$$

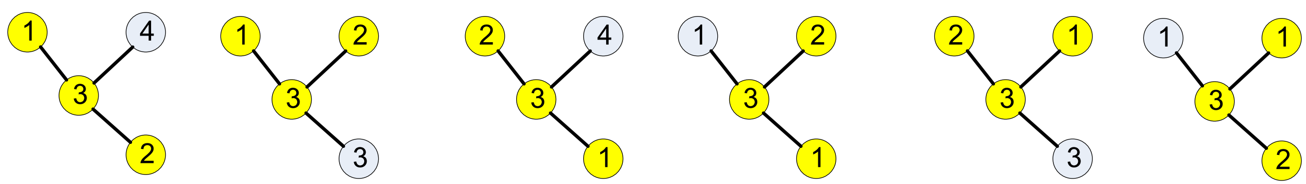
Start- $M^{(0)}$

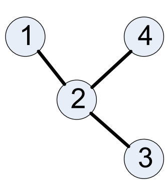
$$\begin{bmatrix}1&0&0&0\\1&1&1&1\\0&1&0&0\end{bmatrix}\quad\begin{bmatrix}0&\boxed{1}&0&0\\1&1&1&1\\0&\boxed{1}&0&0\end{bmatrix}\quad\begin{bmatrix}0&0&1&0\\1&1&1&1\\0&1&0&0\end{bmatrix}\quad\begin{bmatrix}0&0&0&1\\1&1&1&1\\0&1&0&0\end{bmatrix}$$

Innere Knoten: M^0

$$\begin{bmatrix}1&0&0&0\\0&0&1&0\\0&1&0&0\end{bmatrix}\quad\begin{bmatrix}1&0&0&0\\0&0&0&1\\0&1&0&0\end{bmatrix}\quad\begin{bmatrix}0&0&1&0\\1&0&0&0\\0&1&0&0\end{bmatrix}\quad\begin{bmatrix}0&0&1&0\\0&0&0&1\\0&1&0&0\end{bmatrix}\quad\begin{bmatrix}0&0&0&1\\1&0&0&0\\0&1&0&0\end{bmatrix}\quad\begin{bmatrix}0&0&0&1\\0&0&1&0\\0&1&0&0\end{bmatrix}$$

Blätter: P



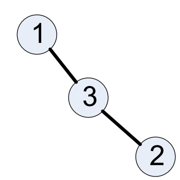


PA_HP^T

$$=\begin{bmatrix}0&0&1\\0&0&1\\1&1&0\end{bmatrix}$$

Verglichen mit A_G

$$=\begin{bmatrix}0&0&1\\0&0&1\\1&1&0\end{bmatrix}$$



Weitere Verfeinerungen des Prunings sind möglich

Take-Home Messages

- Pruning kann anwendungsübergreifend erfolgen
  - Alpha-Beta-Pruning
  - Pure Literal (freie Wahl), Forcierte Werte, Propagierung,
  - Lernen aus Backtracking-Sackgassen
  - Nicht-chronologisches Backtracking
- Pruning kann anwendungsspezifisch erfolgen
  - Subgraph-Isomorphie,
  - Zulässige Heuristik bei A*

- Beste Lösung wird nicht verfehlt