

COMPUTERGESTÜTZTE MUSIKFORSCHUNG 1

Institut für Musikinformatik und Musikwissenschaft
Wintersemester 2025–26



Christophe Weis
christophe.weis@stud.hfm.eu

Woche 07
25.11.2025

Organisation

wöchentlich, Di. 14.30–16.00, K10 Raum 309

Modul Music Processing

- **BA MI (HF)/MW (EF), wiss. Schwerpunkt:** Pflicht (4. Semester)
- **BA MI (HF)/MW (EF), künstl. Schwerpunkt:** Wahlpflicht (6. Semester)
- **BA MW (HF)/MI (EF):** Pflicht (4. Semester) – reduzierter Arbeitsaufwand
- **BA MI/MW (KF):** Pflicht (4. Semester)
- **BA:** Wahlfach

Projektarbeit

- eine selbstständige praktische Arbeit aus den Bereichen Musikkodierung, symbolbasierte Musikverarbeitung und –analyse mit Dokumentation (ca. 5000 Zeichen)

Übungen

- Tutorin: Joanna Friedrich-Sroka
- wöchentlich, Di. 11.15–12.45, K10 Raum 309

14.

Accuracy, Precision, Recall

Accuracy, Precision, Recall

- Metriken zur Evaluierung der Qualität von Messungen, insbesondere von Klassifizierungs-Modellen
- Das im vorherigen Kapitel implementierte Modell zur Erkennung von Fugenthemen-Einsätzen kann als Klassifizierungs-Modell betrachtet werden, da es die verschiedenen Positionen innerhalb der Fugenstimmen entweder als „Themeneinsatz“ oder als „nicht Themeneinsatz“ erkennt.¹
- Alle 3 Metriken können über die Begriffe *True Positive*, *True Negative*, *False Positive*, *False Negative* definiert werden.

¹ Mehr zu Klassifizierungs-Modellen folgt in einem späteren Kapitel!

TP, TN, FP, FN

- Bei einer binären Klassifizierung werden die beiden Target-Klassen oft mit den Labels „1“ und „0“ oder „positiv“ und „negativ“ versehen.
- Bei der Evaluierung eines Klassifizierungs-Modells ist es dementsprechend oft wichtig, zu differenzieren, auf welche Weise das Modell – das heißt bezüglich dieser beiden Klassen – richtig oder falsch liegt.
- Korrekte Vorhersagen:
 - **True Positive (TP):** Bezeichnet den Fall, dass das Modell eine positive Klasse korrekt vorhergesagt hat.
Beispiel: Ein Fugenthemen-Einsatz wurde korrekt erkannt.
 - **True Negative (TN):** Bezeichnet den Fall, dass das Modell eine negative Klasse korrekt vorhergesagt hat.
Beispiel: Eine Position, an der kein Fugenthema einsetzt, wurde korrekt als „kein Fugenthemen-Einsatz“ erkannt.
- Falsche Vorhersagen:
 - **False Positive (FP):** Bezeichnet den Fall, dass das Modell eine positive Klasse falsch vorhergesagt hat.
Beispiel: Eine Position, an der kein Fugenthema einsetzt, wurde als Fugenthemen-Einsatz erkannt.
 - **False Negative (FN):** Bezeichnet den Fall, dass das Modell eine negative Klasse korrekt vorhergesagt hat.
Beispiel: Ein Fugenthemen-Einsatz wurde nicht erkannt.

Accuracy

- Metrik, die misst, wie häufig ein Klassifizierungs-Modell richtig liegt. Hohe Aussagekraft bei gleichmäßig verteilten Klassen.
- *Definition:*

$$\text{Accuracy} := \frac{\text{Korrekte Vorhersagen}}{\text{Alle Vorhersagen}} =$$

- *Beispiel:*

Vorhersage:

x						x					x			x				x					
---	--	--	--	--	--	---	--	--	--	--	---	--	--	---	--	--	--	---	--	--	--	--	--

Ground-Truth:

x					x					x			x				x					
---	--	--	--	--	---	--	--	--	--	---	--	--	---	--	--	--	---	--	--	--	--	--

X = Fugenthemen-Einsatz

$$\text{Accuracy} = \frac{\text{Korrekte Vorhersagen}}{\text{Alle Vorhersagen}} =$$

Accuracy

- Metrik, die misst, wie häufig ein Klassifizierungs-Modell richtig liegt. Hohe Aussagekraft bei gleichmäßig verteilten Klassen.
- *Definition:*

$$\text{Accuracy} := \frac{\text{Korrekte Vorhersagen}}{\text{Alle Vorhersagen}} =$$

- *Beispiel:*

Vorhersage:

x						x					x			x				x					
---	--	--	--	--	--	---	--	--	--	--	---	--	--	---	--	--	--	---	--	--	--	--	--

Ground-Truth:

x					x					x			x				x					
---	--	--	--	--	---	--	--	--	--	---	--	--	---	--	--	--	---	--	--	--	--	--

X = Fugenthemen-Einsatz

$$\text{Accuracy} = \frac{\text{Korrekte Vorhersagen}}{\text{Alle Vorhersagen}} = \frac{28}{30} \cong 0.93$$

Accuracy

- Metrik, die misst, wie häufig ein Klassifizierungs-Modell richtig liegt. Hohe Aussagekraft bei gleichmäßig verteilten Klassen.
- *Definition:*

$$\text{Accuracy} := \frac{\text{Korrekte Vorhersagen}}{\text{Alle Vorhersagen}} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

- *Beispiel:*

Vorhersage:

x						x					x			x				x					
---	--	--	--	--	--	---	--	--	--	--	---	--	--	---	--	--	--	---	--	--	--	--	--

Ground-Truth:

x					x					x			x				x					
---	--	--	--	--	---	--	--	--	--	---	--	--	---	--	--	--	---	--	--	--	--	--

X = Fugenthemen-Einsatz

$$\text{Accuracy} = \frac{\text{Korrekte Vorhersagen}}{\text{Alle Vorhersagen}} = \frac{28}{30} \cong 0.93$$

Accuracy

- *Nachteil:*
Bei **ungleichmäßig verteilten Klassen** hat der Wert alleine wenig Aussagekraft über die tatsächliche Qualität des Modells.
 - *Beispiel:*

Vorhersage:



Ground-Truth:



X = Fugenthemen-Einsatz

$$\text{Accuracy} = \frac{\text{Korrekte Vorhersagen}}{\text{Alle Vorhersagen}} = \frac{25}{30} \cong 0.83$$

→ verhältnismäßig hoher Accuracy-Wert, obwohl kein einziger Themen-Einsatz vorhergesagt wurde.

Precision

- Metrik, die misst, wie häufig ein Klassifizierungs-Modell die „positive“ Klasse richtig vorhersagt. Hohe Aussagekraft bei ungleichmäßig verteilten Klassen und wenn „False Positives“ unerwünscht sind.
- *Definition:*

$$\text{Precision} := \frac{\text{TP}}{\text{TP} + \text{FP}}$$

- *Beispiel:*

Vorhersage:

x			x		x	x		x		x	x	x	x	x	x	x	x	x		
---	--	--	---	--	---	---	--	---	--	---	---	---	---	---	---	---	---	---	--	--

Ground-Truth:

x					x				x			x			x			x	
---	--	--	--	--	---	--	--	--	---	--	--	---	--	--	---	--	--	---	--

x = Fugenthemen-Einsatz

Precision =

Precision

- Metrik, die misst, wie häufig ein Klassifizierungs-Modell die „positive“ Klasse richtig vorhersagt. Hohe Aussagekraft bei ungleichmäßig verteilten Klassen und wenn „False Positives“ unerwünscht sind.
- *Definition:*

$$\text{Precision} := \frac{\text{TP}}{\text{TP} + \text{FP}}$$

- *Beispiel:*

Vorhersage:

x			x		x	x		x		x	x	x	x	x	x	x	x	x	x		
---	--	--	---	--	---	---	--	---	--	---	---	---	---	---	---	---	---	---	---	--	--

Ground-Truth:

x					x				x			x			x			x			
---	--	--	--	--	---	--	--	--	---	--	--	---	--	--	---	--	--	---	--	--	--

x = Fugenthemen-Einsatz

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{5}{10} = 0.5 \quad (\text{zum Vergleich: Accuracy hier } \cong 0.83)$$

Precision

- *Nachteil:*
Berücksichtigt nicht die „False Negatives“, also verpasste Targets.
 - *Beispiel:*

Vorhersage:



Ground-Truth:



$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{1}{1} = 1.00$$

X = Fugenthemen-Einsatz

→ perfekter Precision-Wert, obwohl die meisten Themen-Einsätze nicht erkannt wurden.

Recall

- Metrik, die misst, wie häufig ein Klassifizierungs-Modell die „positive“ Klasse richtig vorhersagt. Hohe Aussagekraft bei ungleichmäßig verteilten Klassen und wenn „False Negatives“ unerwünscht sind.
- *Definition:*

$$\text{Recall} := \frac{\text{TP}}{\text{TP} + \text{FN}}$$

- *Beispiel:*

Vorhersage:



Ground-Truth:



X = Fugenthemen-Einsatz

Recall =

Recall

- Metrik, die misst, wie häufig ein Klassifizierungs-Modell die „positive“ Klasse richtig vorhersagt. Hohe Aussagekraft bei ungleichmäßig verteilten Klassen und wenn „False Negatives“ unerwünscht sind.
- *Definition:*

$$\text{Recall} := \frac{\text{TP}}{\text{TP} + \text{FN}}$$

- *Beispiel:*

Vorhersage:



Ground-Truth:



X = Fugenthemen-Einsatz

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{2}{5} = 0.4 \quad (\text{zum Vergleich: Precision hier} = 1.00)$$

Recall

- *Nachteil:*
Berücksichtigt nicht die „False Positives“.
 - *Beispiel:*

Vorhersage:

Ground-Truth:

X **X** **X** **X** **X**

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{5}{5} = 1.00$$

X = Fugenthemen-Einsatz

→ perfekter Recall-Wert, obwohl alle Positionen als Themen-Einsatz vorhergesagt wurden.

15.

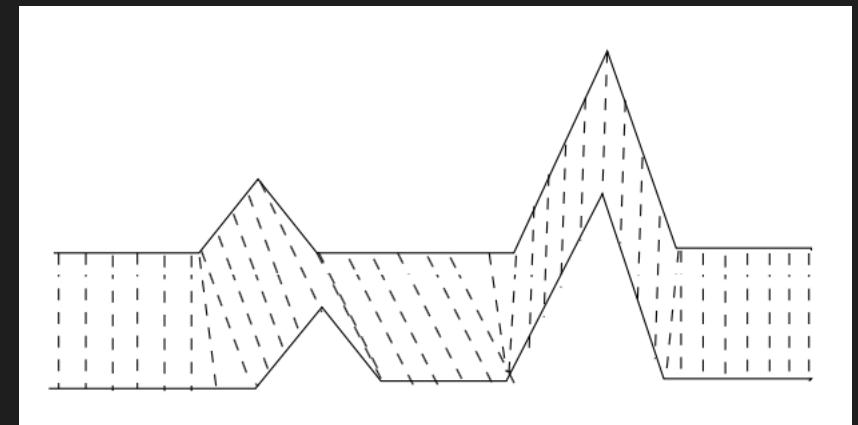
Alignment

Alignment

- *Ziel:*
 - Erkennung von übereinstimmenden Bereichen oder Bereichen hoher Ähnlichkeit zwischen 2 Sequenzen
 - Verknüpfung oder Synchronisierung dieser Bereiche miteinander, bzw. Bestimmung einer Abbildung dieser Bereiche aufeinander
- **Globale Alignments** → Suche nach Übereinstimmungen entlang der vollständigen Länge von 2 Sequenzen
- **Lokale Alignments** → Fokus auf lokale Übereinstimmungen innerhalb von (möglicherweise) insgesamt voneinander abweichenden Sequenzen
- **Offline-Szenarios:**
 - Zu vergleichende Sequenzen sind vor dem Alignment vollständig bekannt
 - *Beispiel:*
 - Erkennung von Coverversionen von Songs
- **Online-Szenarios:**
 - Alignment erfolgt schrittweise, mit fortlaufender Verfügbarkeit neuer Informationen
 - *Beispiele:*
 - Score-Following
 - Automatische Generierung von Begleitungen

Beispiele für Algorithmen

- **Needleman-Wunsch-Algorithmus:**
 - Bestimmung eines (oder mehrerer) **globalen optimalen Alignments** zwischen 2 Sequenzen
 - Anwendungen in der **Bioinformatik** zum Vergleichen von Protein- oder DNA-Sequenzen
 - Nutzung in der computergestützten Musikforschung, z. B. zum Alignment von Melodien
- **Smith-Waterman-Algorithmus:**
 - Bestimmung des **lokalen optimalen Alignments** zwischen 2 Sequenzen
 - Ähnliche Anwendungsbereiche wie der Needleman-Wunsch-Algorithmus
- **Dynamic-Time-Warping (siehe späteres Kapitel zu DTW):**
 - Messung der Ähnlichkeit zwischen 2 zeitlichen Sequenzen
 - Zu vergleichende Sequenzen können asynchron sein, z. B. unterschiedliche musikalische Tempi besitzen
 - *Ziel:*
Jeder zeitliche Index der ersten Sequenz soll mit einem (oder mehreren) entsprechenden Index der zweiten Sequenz gematcht werden (und umgekehrt)
- Anwendungsbeispiele:
<http://rna.informatik.uni-freiburg.de/Teaching/index.jsp?toolName=Needleman-Wunsch>



https://en.wikipedia.org/wiki/Dynamic_time_warping#/media/File:Dynamic_time_warping.png

Needleman-Wunsch

- Algorithmus zur Bestimmung eines **globalen optimalen Alignments** zwischen 2 Sequenzen.
- Grundlage: **Scores von Matches, Mismatches und Gaps** zwischen den Sequenzen.
- Berechnung einer Score-Matrix:
 - Bestimmung des optimalen **Scores für alle Präfixe** der beiden Sequenzen.
 - Der final erhaltene Alignment-Score steht in der unteren rechten Ecke in der Matrix.
Der Alignment-Score ist ein **Maß für die Ähnlichkeit** der beiden Sequenzen.
 - Optimales globales **Alignment durch Backtracking**: Rückverfolgung aller optimalen Pfade innerhalb der Matrix.
- Der Algorithmus ist effizient, kann bei sehr langen Sequenzen rechenintensiv werden.
- Beispiel eines mit dem Needleman-Wunsch-Algorithmus bestimmten globalen optimalen Alignments:

Vorgegebene Scores:

Match: 1

Mismatch: -1

Gap: -2

Ausgangssequenzen:

Sequenz 1: ACGTA

Sequenz 2: AGTC

Alignment:

ACGTA

| | | x

A - GTC

Score des globalen optimalen Alignments:

... ?

Needleman-Wunsch

- Algorithmus zur Bestimmung eines **globalen optimalen Alignments** zwischen 2 Sequenzen.
- Grundlage: **Scores von Matches, Mismatches und Gaps** zwischen den Sequenzen.
- Berechnung einer Score-Matrix:
 - Bestimmung des optimalen **Scores für alle Präfixe** der beiden Sequenzen.
 - Der final erhaltene Alignment-Score steht in der unteren rechten Ecke in der Matrix.
Der Alignment-Score ist ein **Maß für die Ähnlichkeit** der beiden Sequenzen.
 - Optimales globales **Alignment durch Backtracking**: Rückverfolgung aller optimalen Pfade innerhalb der Matrix.
- Der Algorithmus ist effizient, kann bei sehr langen Sequenzen rechenintensiv werden.
- Beispiel eines mit dem Needleman-Wunsch-Algorithmus bestimmten globalen optimalen Alignments:

Vorgegebene Scores:

Match: 1

Mismatch: -1

Gap: -2

Ausgangssequenzen:

Sequenz 1: ACGTA

Sequenz 2: AGTC

Alignment:

ACGTA

| | | x

A - GTC

Score des globalen optimalen Alignments:

$1 + (-2) + 1 + 1 + (-1) = 0$

Needleman-Wunsch

- Bestimmung des optimalen Alignments von 2 Sequenzen a und b mit Länge $\text{len}(a) = m$ und $\text{len}(b) = n$ durch Matrix-Rekursion
 - Bestimmung des optimalen Alignments und eines Alignment-Scores für alle möglich Präfixe (= Sequenzen-Anfänge)
 - Der Score $S(i, j)$ bewertet das Alignment der Präfixe $a_1 a_2 \dots a_i$ und $b_1 b_2 \dots b_j$

- $S(0, 0) = 0$

- $S(i, 0) = S(i - 1, 0) + \text{Gap-Kosten}$ für $1 \leq i \leq m$

- $S(0, j) = S(0, j - 1) + \text{Gap-Kosten}$ für $1 \leq j \leq n$

- $S(i, j) = \max \begin{cases} S(i - 1, j - 1) + \text{Match-Score} & \text{falls } a_i = b_j \\ S(i - 1, j - 1) + \text{Mismatch-Kosten} & \text{falls } a_i \neq b_j \\ S(i, j - 1) + \text{Gap-Kosten} & \text{falls } a_j = - \\ S(i - 1, j) + \text{Gap-Kosten} & \text{falls } b_j = - \end{cases}$

Needleman-Wunsch

- *Beispiel:*

Berechnung des Alignment-Scores und Bestimmung eines Alignments der Sequenzen

$a = \text{ACGTA}$ und $b = \text{AGTC}$

	-	A	C	G	T	A
-						
A						
G						
T						
C						

Vorgegebene Scores:

Match: 1

Mismatch: -1

Gap: -2

Needleman-Wunsch

- *Beispiel:*

Berechnung des Alignment-Scores und Bestimmung eines Alignments der Sequenzen

$a = \text{ACGTA}$ und $b = \text{AGTC}$

	-	A	C	G	T	A
-	0	-2	-4	-6	-8	-10
A	-2					
G	-4					
T	-6					
C	-8					

Schritt 1:

Initialisierung der **1. Zeile** und **1. Spalte** der Matrix

Vorgegebene Scores:

Match: 1

Mismatch: -1

Gap: -2

Needleman-Wunsch

- Beispiel:

Berechnung des Alignment-Scores und Bestimmung eines Alignments der Sequenzen

$a = \text{ACGTA}$ und $b = \text{AGTC}$

	-	A	C	G	T	A
-	0	-2	-4	-6	-8	-10
A	-2	1	-1	-3	-5	-7
G	-4	-1	?			
T	-6	-3				
C	-8	-5				

Vorgegebene Scores:

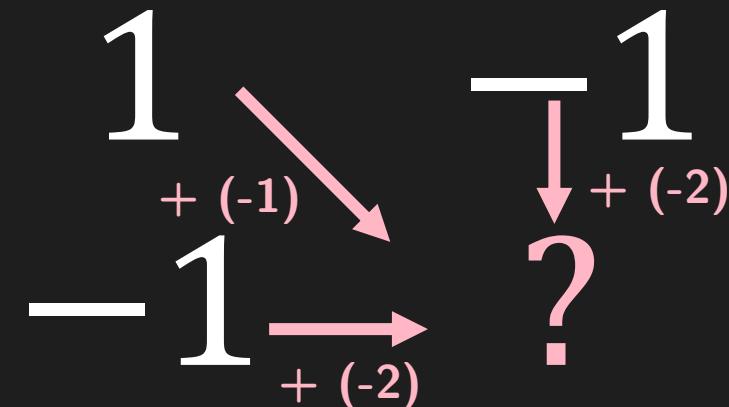
Match: 1

Mismatch: -1

Gap: -2

Ausfüllen aller Matrix-Einträge:
Bestimmung des Maximums aus

- oberer Wert + Gap-Kosten
- linker Wert + Gap-Kosten
- diagonaler Wert + Match-Score / Mismatch-Kosten



Needleman-Wunsch

- Beispiel:

Berechnung des Alignment-Scores und Bestimmung eines Alignments der Sequenzen

$a = \text{ACGTA}$ und $b = \text{AGTC}$

	-	A	C	G	T	A
-	0	-2	-4	-6	-8	-10
A	-2	1	-1	-3	-5	-7
G	-4	-1	0			
T	-6	-3				
C	-8	-5				

Vorgegebene Scores:

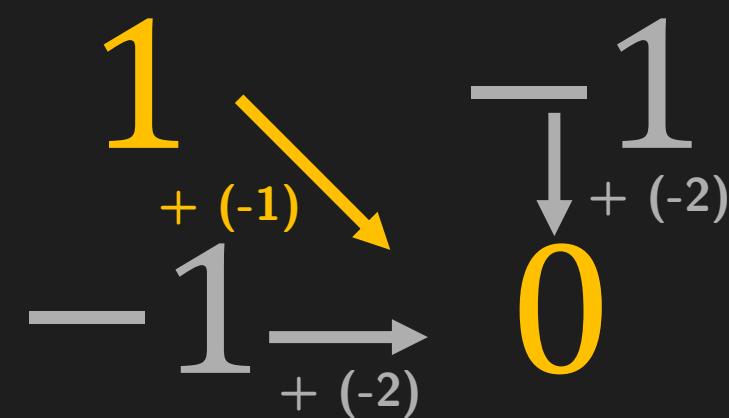
Match: 1

Mismatch: -1

Gap: -2

Ausfüllen aller Matrix-Einträge:
Bestimmung des Maximums aus

- oberer Wert + Gap-Kosten
- linker Wert + Gap-Kosten
- diagonaler Wert + Match-Score / Mismatch-Kosten



Needleman-Wunsch

- *Beispiel:*

Berechnung des Alignment-Scores und Bestimmung eines Alignments der Sequenzen

$a = \text{ACGTA}$ und $b = \text{AGTC}$

	-	A	C	G	T	A
-	0	-2	-4	-6	-8	-10
A	-2	1	-1	-3	-5	-7
G	-4	-1	0	0	-2	-4
T	-6	-3	-2	-1	1	-1
C	-8	-5	-2	-3	-1	0

Ausfüllen aller Matrix-Einträge:

... bis die **Scores aller Präfixes** berechnet wurden

Vorgegebene Scores:

Match: 1

Mismatch: -1

Gap: -2

Needleman-Wunsch

- *Beispiel:*

Berechnung des Alignment-Scores und Bestimmung eines Alignments der Sequenzen

$a = \text{ACGTA}$ und $b = \text{AGTC}$

	-	A	C	G	T	A
-	0	-2	-4	-6	-8	-10
A	-2	1	-1	-3	-5	-7
G	-4	-1	0	0	-2	-4
T	-6	-3	-2	-1	1	-1
C	-8	-5	-2	-3	-1	0

Vorgegebene Scores:

Match: 1

Mismatch: -1

Gap: -2

Letzter Schritt:

Bestimmung des optimalen globalen Alignments durch **Backtracking**:

- Start: Eintrag unten rechts
- Ziel: Eintrag oben links
- Zu jedem Zeitpunkt:
Auswahl des **Pfades**, über den sich der **aktuelle Wert berechnen lässt**
(mehrere Pfade sind möglich)

Needleman-Wunsch

- Beispiel:

Berechnung des Alignment-Scores und Bestimmung eines Alignments der Sequenzen

$a = \text{ACGTA}$ und $b = \text{AGTC}$

	-	A	C	G	T	A
-	0	-2	-4	-6	-8	-10
A	-2	1	-1	-3	-5	-7
G	-4	-1	0	0	-2	-4
T	-6	-3	-2	-1	1	-1
C	-8	-5	-2	-3	-1	0

Vorgegebene Scores:

Match: 1

Mismatch: -1

Gap: -2

Alignment:

ACGTA

| | | x

A-GTC

Score des Alignments:

0

Letzter Schritt:

Bestimmung des optimalen globalen Alignments durch **Backtracking**:

- Start: Eintrag unten rechts
- Ziel: Eintrag oben links
- Zu jedem Zeitpunkt:
Auswahl des **Pfades**, über den sich der **aktuelle Wert berechnen lässt**
(mehrere Pfade sind möglich)

Smith-Waterman

- Algorithmus zur Bestimmung eines **lokalen optimalen Alignments** zwischen 2 Sequenzen.
- 2 Änderungen im Vergleich zum Needleman-Wunsch-Algorithmus:
 - Initialisierung der 1. Zeile und der 1. Spalte mit 0
 - Während der Rekursion: Maximierung über einen zusätzlichen Wert: 0
- Der final erhaltene lokale Alignment-Score ist der Matrix-Eintrag mit dem größten Wert (und steht nicht mehr unbedingt in der unteren rechten Ecke).
- Bestimmung eines lokalen optimalen **Alignments durch Backtracking**, beginnend mit dem (oder einem) Eintrag mit dem größten Wert bis zu einem Eintrag mit dem Wert 0.

