

# *Syntax natürlicher Sprachen*

Tutorium 05:

Dependenzgrammatik & Dependency Parsing

Shuyan Liu

22.11.2024

Einige Beispiele kommt aus der Vorlesungsfolien, Aufgaben sowie Übungen.

Die Hauptteile der Slides dieser Woche stammen von Sarah Anna Uffermann  
aus Wintersemester 2023/24 und wurden bearbeitet. Verwendung mit Dank.

# Nachtrag zur letzten Sitzung

- Aufgabeblatt 4: Recursive Descent Parser:
  - Funktioniert, weil die NP-Regeln rechtsrekursiv sind
- Aufgabeblatt 4: Earley Parser (manuell)

# Nachtrag zur letzten Sitzung

## Aufgabeblatt 4: Earley Parser (manuell)

$S \rightarrow NP VP$

$NP \rightarrow DET NOM$

$NOM \rightarrow ADJ NOM \mid N$

$VP \rightarrow V NP \mid V$

$ADJ \rightarrow \text{"kluge"} \mid \text{"dicke"}$

$DET \rightarrow \text{"der"} \mid \text{"die"} \mid \text{"das"}$

$N \rightarrow \text{"Frau"} \mid \text{"Buch"}$

$V \rightarrow \text{"kennt"}$

Scan Predict Complete

|  | die  | kluge  | Frau   | kennt   | das  | dicke  | Buch  |  |
|--|--|--|--|---|--|--|---|--|
| $S \rightarrow NP VP$<br>$NP \rightarrow \cdot DET NOM$<br>$DET \rightarrow \cdot der$<br>$DET \rightarrow \cdot die$<br>$DET \rightarrow \cdot das$ | $DET \rightarrow \cdot die$<br>$NP \rightarrow DET \cdot NOM$  |  | $NP \rightarrow DET \cdot NOM$<br>$S \rightarrow NP \cdot VP$                    | $S \rightarrow NP VP \cdot$   |  |  | $S \rightarrow NP VP \cdot$                             |  |
|  | $NOM \rightarrow \cdot ADJ NOM$<br>$NOM \rightarrow \cdot N$<br>$ADJ \rightarrow \cdot kluge$<br>$ADJ \rightarrow \cdot dicke$<br>$N \rightarrow \cdot Frau$<br>$N \rightarrow \cdot Buch$ | $ADJ \rightarrow kluge \cdot$<br>$NOM \rightarrow ADJ \cdot NOM$   | $NOM \rightarrow ADJ NOM \cdot$  |   |  |  |   |  |
|  |  | $NOM \rightarrow \cdot ADJ NOM$<br>$NOM \rightarrow \cdot N$<br>$ADJ \rightarrow \cdot kluge$<br>$ADJ \rightarrow \cdot dicke$<br>$N \rightarrow \cdot Frau$<br>$N \rightarrow \cdot Buch$ | $N \rightarrow Frau \cdot$<br>$NOM \rightarrow N \cdot$                          |   |  |  |   |  |
|  |  |  | $VP \rightarrow \cdot V NP$<br>$VP \rightarrow \cdot V$<br>$V \rightarrow kennt$ | $V \rightarrow kennt \cdot$<br>$VP \rightarrow V \cdot NP$<br>$VP \rightarrow V \cdot$                                      |  |  | $VP \rightarrow V NP \cdot$                             |  |
|  |  |  |  | $NP \rightarrow \cdot DET NOM$<br>$DET \rightarrow \cdot der$<br>$DET \rightarrow \cdot die$<br>$DET \rightarrow \cdot das$ | $DET \rightarrow das \cdot$<br>$NP \rightarrow DET \cdot NOM$  |  | $NP \rightarrow DET NOM \cdot$                          |  |
|  |  |  |  |   | $NOM \rightarrow \cdot ADJ NOM$<br>$NOM \rightarrow \cdot N$<br>$ADJ \rightarrow \cdot kluge$<br>$ADJ \rightarrow \cdot dicke$<br>$N \rightarrow \cdot Frau$<br>$N \rightarrow \cdot Buch$ | $ADJ \rightarrow dicke \cdot$<br>$NOM \rightarrow ADJ \cdot NOM$   | $NOM \rightarrow ADJ NOM \cdot$                         |  |
|  |  |  |  |   |  | $NOM \rightarrow \cdot ADJ NOM$<br>$NOM \rightarrow \cdot N$<br>$ADJ \rightarrow \cdot kluge$<br>$ADJ \rightarrow \cdot dicke$<br>$N \rightarrow \cdot Frau$<br>$N \rightarrow \cdot Buch$ | $N \rightarrow Buch \cdot$<br>$NOM \rightarrow N \cdot$ |  |

## 2 Ansätze zur Syntaktischen Analyse

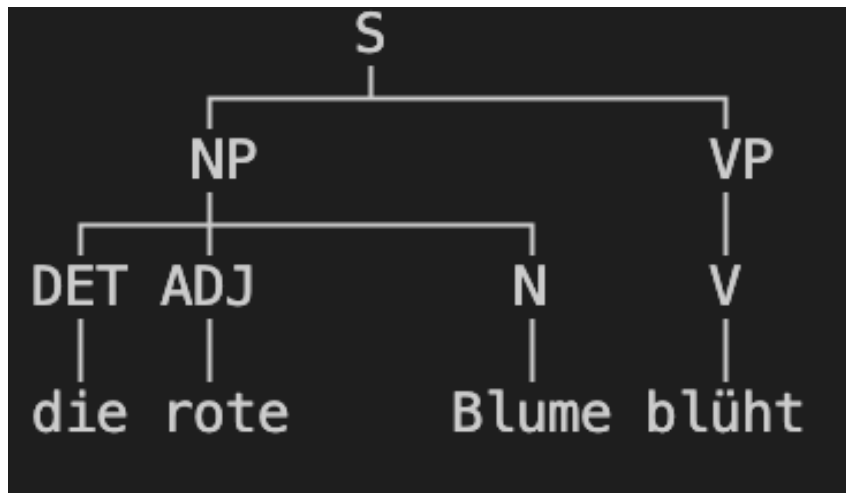
- Konstituentenstruktur:
  - Wie kann man ein Satz in syntaktischen Einheiten (Konstituente) zerlegen?
  - Untersucht **Strukturregeln** zur Erklärung des Aufbaus eines Satzes
- Dependenz:
  - In welcher syntaktischen Beziehung stehen **Wörter** in einem Satz?
  - Untersucht Abhängigkeitsverhältnisse zwischen Wörtern eines Satzes

# Dependenzrelation

- Dependent:
  - $y$  hängt von  $x$  ab  $\rightarrow y$  ist **DEPENDENT** von  $x$
  - $x$  regiert  $y \rightarrow x$  ist **KOPF** von  $y$
- Beispiel:
  - Satz: der Mann rennt
  - „der“ ist Dependent von „Mann“
  - „Mann“ ist Kopf von „der“

# Konstituenz vs. Dependenz

## Konstituenz



S → NP VP

NP → Det ADJ N

VP → V

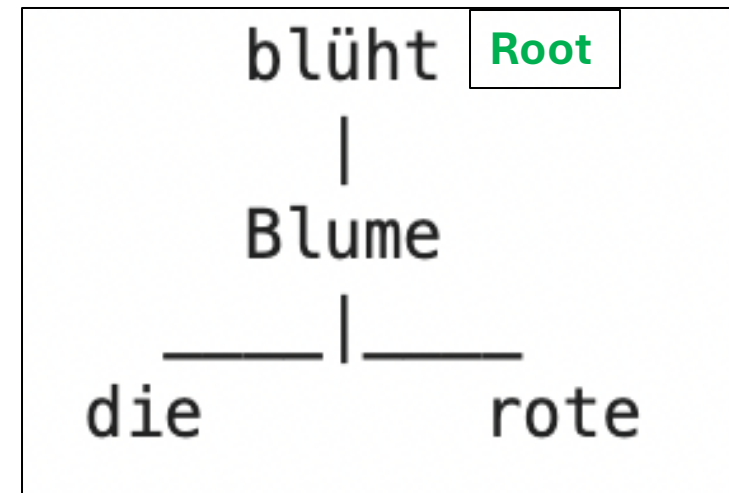
Det → "die"

N → "Blume"

ADJ → "rote"

V → "blüht"

## Dependenz



Darstellung  
mit NLTK

**Kopf → Dependent**

"blüht" → „Blume“

"Blume" → "die" | "rote"

# Rektion und Modifikation

- Rektion:

- Kopf **kann nicht** ohne Dependent auftreten (und ein Dependent kann nie ohne Kopf auftreten)
- Bilaterale Dependenz: Kopf und Dependent sind voneinander abhängig
- Dependent ist **Komplement** des Kopfes

„Sie betrachtet das Bild.“

\* „Sie betrachtet.“ **Nicht wohlgeformt**

\* „Sie das Bild.“ **Nicht wohlgeformt**

- Modifikation

- Kopf **kann** ohne Dependent auftreten
- Unilaterale Dependenz
- Dependent ist **Adjunkt** des Kopfes

„Sie betrachtet das Bild am Nachmittag.“

„Sie betrachtet das Bild.“ **Wohlgeformt**

# Arten von Dependents

- Obligatorischer Dependent
  - Komplement
- Fakultativer Dependent
  - Komplement, aber nach Kontext weglassbar
  - Klassisches Beispiel: „Er sieht das Huhn“ → „Er sieht“
- Optionaler Dependent
  - Adjunkt (zusätzliche Information)



# Primacy of Content Words

- In Konstituentenstruktur
  - X ist Kopf einer XP
  - Beispiel: P ist Kopf einer PP

Aber...

- Bei Dependenz
  - Nur **Inhaltswörter** sind Köpfe
  - Beispiel: „an der Uni“ ist Kopf von im (UD-Schema)

# Shift-Reduce Dependency Parsing

- Ein Satz ist eine Liste (Buffer) von Wörtern
- Der Parser parst Wort für Wort
- Ein Stack für Wörter und Strukturen
- Operationen
  - **Shift**: Schiebe ein Wort aus der Liste auf das Stack
  - **Reduce**: Relation zwischen den beiden obersten Elementen auf das Stack hinzufügen und den Dependents vom Stack löschen
    - **LEFTARC**: Kopf legt rechts vom Dependent
    - **RIGHTARC**: Kopf legt links vom Dependent

ein Beispiel  
macht Sinn



# Shift-Reduce Dependency Parsing

## Beginn:

- Alle Wörter sind in der Liste (Buffer)
- Stack ist mit ROOT initialisiert
- Es sind noch keine Pfeile (Dependenzrelationen) vorhanden
- Erste Operation: SHIFT

**REDUCE-LEFTARC:** ist immer möglich

**REDUCE-RIGHTARC:** nur dann möglich, **wenn der Dependent der Relation selbst nicht Kopf einer der noch offenen Relation ist.**

Diese Einschränkung verhindert, dass ein Wort zu früh vom Stack genommen wird.

Beispiele kommen gleich

# Shift-Reduce Dependency Parsing

Satz: „Das Mädchen sieht das Huhn.“

Buffer: [Das, Mädchen, sieht, das, Huhn]

Stack: [ROOT]

| <u>Buffer</u>                    | <u>Stack</u> | <u>Operation</u> |
|----------------------------------|--------------|------------------|
| [Das, Mädchen, sieht, das, Huhn] | [ROOT]       | <b>SHIFT</b>     |

Das Mädchen sieht das Huhn.

# Shift-Reduce Dependency Parsing

| Buffer                           | Stack       | Operation    |
|----------------------------------|-------------|--------------|
| [Das, Mädchen, sieht, das, Huhn] | [ROOT]      | SHIFT        |
| [Mädchen, sieht, das, Huhn]      | [ROOT, Das] | <b>SHIFT</b> |

Das Mädchen sieht das Huhn.

# Shift-Reduce Dependency Parsing

| Buffer                           | Stack                | Operation |
|----------------------------------|----------------------|-----------|
| [Das, Mädchen, sieht, das, Huhn] | [ROOT]               | SHIFT     |
| [Mädchen, sieht, das, Huhn]      | [ROOT, Das]          | SHIFT     |
| [sieht, das, Huhn]               | [ROOT, Das, Mädchen] | LEFTARC   |



Das Mädchen sieht das Huhn.

# Shift-Reduce Dependency Parsing

| Buffer                           | Stack                | Operation |
|----------------------------------|----------------------|-----------|
| [Das, Mädchen, sieht, das, Huhn] | [ROOT]               | SHIFT     |
| [Mädchen, sieht, das, Huhn]      | [ROOT, Das]          | SHIFT     |
| [sieht, das, Huhn]               | [ROOT, Das, Mädchen] | LEFTARC   |
| [sieht, das, Huhn]               | [ROOT, Mädchen]      | SHIFT     |



Das Mädchen sieht das Huhn.

# Shift-Reduce Dependency Parsing

| Buffer                           | Stack                  | Operation |
|----------------------------------|------------------------|-----------|
| [Das, Mädchen, sieht, das, Huhn] | [ROOT]                 | SHIFT     |
| [Mädchen, sieht, das, Huhn]      | [ROOT, Das]            | SHIFT     |
| [sieht, das, Huhn]               | [ROOT, Das, Mädchen]   | LEFTARC   |
| [sieht, das, Huhn]               | [ROOT, Mädchen] SHIFT  |           |
| [das, Huhn]                      | [ROOT, Mädchen, sieht] | LEFTARC   |



Das Mädchen sieht das Huhn.



# Shift-Reduce Dependency Parsing

| Buffer                           | Stack                  | Operation |
|----------------------------------|------------------------|-----------|
| [Das, Mädchen, sieht, das, Huhn] | [ROOT]                 | SHIFT     |
| [Mädchen, sieht, das, Huhn]      | [ROOT, Das]            | SHIFT     |
| [sieht, das, Huhn]               | [ROOT, Das, Mädchen]   | LEFTARC   |
| [sieht, das, Huhn]               | [ROOT, Mädchen] SHIFT  |           |
| [das, Huhn]                      | [ROOT, Mädchen, sieht] | LEFTARC   |
| [das, Huhn]                      | [ROOT, sieht]          | SHIFT     |



Das Mädchen sieht das Huhn.

# Shift-Reduce Dependency Parsing

| Buffer                           | Stack                  | Operation |
|----------------------------------|------------------------|-----------|
| [Das, Mädchen, sieht, das, Huhn] | [ROOT]                 | SHIFT     |
| [Mädchen, sieht, das, Huhn]      | [ROOT, Das]            | SHIFT     |
| [sieht, das, Huhn]               | [ROOT, Das, Mädchen]   | LEFTARC   |
| [sieht, das, Huhn]               | [ROOT, Mädchen] SHIFT  |           |
| [das, Huhn]                      | [ROOT, Mädchen, sieht] | LEFTARC   |
| [das, Huhn]                      | [ROOT, sieht]          | SHIFT     |
| [Huhn]                           | [ROOT, sieht, das]     | SHIFT     |



Das Mädchen sieht das Huhn.

# Shift-Reduce Dependency Parsing

| Buffer                           | Stack                      | Operation      |
|----------------------------------|----------------------------|----------------|
| [Das, Mädchen, sieht, das, Huhn] | [ROOT]                     | SHIFT          |
| [Mädchen, sieht, das, Huhn]      | [ROOT, Das]                | SHIFT          |
| [sieht, das, Huhn]               | [ROOT, Das, Mädchen]       | LEFTARC        |
| [sieht, das, Huhn]               | [ROOT, Mädchen]      SHIFT |                |
| [das, Huhn]                      | [ROOT, Mädchen, sieht]     | LEFTARC        |
| [das, Huhn]                      | [ROOT, sieht]      SHIFT   |                |
| [Huhn]                           | [ROOT, sieht, das]         | SHIFT          |
| [ ]                              | [ROOT, sieht, das, Huhn]   | <b>LEFTARC</b> |



Das Mädchen sieht das Huhn.

# Shift-Reduce Dependency Parsing

| Buffer                           | Stack                      | Operation       |
|----------------------------------|----------------------------|-----------------|
| [Das, Mädchen, sieht, das, Huhn] | [ROOT]                     | SHIFT           |
| [Mädchen, sieht, das, Huhn]      | [ROOT, Das]                | SHIFT           |
| [sieht, das, Huhn]               | [ROOT, Das, Mädchen]       | LEFTARC         |
| [sieht, das, Huhn]               | [ROOT, Mädchen]      SHIFT |                 |
| [das, Huhn]                      | [ROOT, Mädchen, sieht]     | LEFTARC         |
| [das, Huhn]                      | [ROOT, sieht]      SHIFT   |                 |
| [Huhn]                           | [ROOT, sieht, das]         | SHIFT           |
| [ ]                              | [ROOT, sieht, das, Huhn]   | LEFTARC         |
| [ ]                              | [ROOT, sieht, Huhn]        | <b>RIGHTARC</b> |



Das Mädchen sieht das Huhn.

# Shift-Reduce Dependency Parsing

| Buffer                           | Stack                      | Operation       |
|----------------------------------|----------------------------|-----------------|
| [Das, Mädchen, sieht, das, Huhn] | [ROOT]                     | SHIFT           |
| [Mädchen, sieht, das, Huhn]      | [ROOT, Das]                | SHIFT           |
| [sieht, das, Huhn]               | [ROOT, Das, Mädchen]       | LEFTARC         |
| [sieht, das, Huhn]               | [ROOT, Mädchen]      SHIFT |                 |
| [das, Huhn]                      | [ROOT, Mädchen, sieht]     | LEFTARC         |
| [das, Huhn]                      | [ROOT, sieht]              | SHIFT           |
| [Huhn]                           | [ROOT, sieht, das]         | SHIFT           |
| [ ]                              | [ROOT, sieht, das, Huhn]   | LEFTARC         |
| [ ]                              | [ROOT, sieht, Huhn]        | RIGHTARC        |
| [ ]                              | [ROOT, sieht]              | <b>RIGHTARC</b> |



Das Mädchen sieht das Huhn.

**ROOT**

# Shift-Reduce Dependency Parsing

| Buffer                           | Stack                      | Operation   |
|----------------------------------|----------------------------|-------------|
| [Das, Mädchen, sieht, das, Huhn] | [ROOT]                     | SHIFT       |
| [Mädchen, sieht, das, Huhn]      | [ROOT, Das]                | SHIFT       |
| [sieht, das, Huhn]               | [ROOT, Das, Mädchen]       | LEFTARC     |
| [sieht, das, Huhn]               | [ROOT, Mädchen]      SHIFT |             |
| [das, Huhn]                      | [ROOT, Mädchen, sieht]     | LEFTARC     |
| [das, Huhn]                      | [ROOT, sieht]              | SHIFT       |
| [Huhn]                           | [ROOT, sieht, das]         | SHIFT       |
| [ ]                              | [ROOT, sieht, das, Huhn]   | LEFTARC     |
| [ ]                              | [ROOT, sieht, Huhn]        | RIGHTARC    |
| [ ]                              | [ROOT, sieht]              | RIGHTARC    |
| [ ]                              | [ROOT]                     | <b>DONE</b> |



Das Mädchen sieht das Huhn.

**ROOT**

# Shift-Reduce Dependency Parsing

Satz: „Kauf Tickets nach München.“

Buffer: [Kauf, Tickets, nach, München]

Stack: [ROOT]

| <u>Buffer</u>                  | <u>Stack</u> | <u>Operation</u> |
|--------------------------------|--------------|------------------|
| [Kauf, Tickets, nach, München] | [ROOT]       | <b>SHIFT</b>     |

Kauf Tickets nach München.

# Shift-Reduce Dependency Parsing

| <u>Buffer</u>                  | <u>Stack</u> | <u>Operation</u> |
|--------------------------------|--------------|------------------|
| [Kauf, Tickets, nach, München] | [ROOT]       | SHIFT            |
| [Tickets, nach, München]       | [ROOT, Kauf] |                  |

Welche Operation ist jetzt die richtige?

Kauf Tickets nach München.



# Shift-Reduce Dependency Parsing

| Buffer                         | Stack        | Operation |
|--------------------------------|--------------|-----------|
| [Kauf, Tickets, nach, München] | [ROOT]       | SHIFT     |
| [Tickets, nach, München]       | [ROOT, Kauf] |           |

Möglichkeiten:

- Rightarc: „Kauf“ wird als Root markiert und vom Stack genommen.
- Shift: Wir legen das nächste Wort auf den Stack.

Richtig ist **SHIFT**, da das Parsen des restlichen Satzes nicht mehr möglich wäre, wenn wir „Kauf“ vom Stack nehmen.

Kauf Tickets nach München.

# Shift-Reduce Dependency Parsing

| <u>Buffer</u>                  | <u>Stack</u> | <u>Operation</u> |
|--------------------------------|--------------|------------------|
| [Kauf, Tickets, nach, München] | [ROOT]       | SHIFT            |
| [Tickets, nach, München]       | [ROOT, Kauf] | <b>SHIFT</b>     |

Kauf Tickets nach München.

| Buffer                         | Stack                 | Operation |
|--------------------------------|-----------------------|-----------|
| [Kauf, Tickets, nach, München] | [ROOT]                | SHIFT     |
| [Tickets, nach, München]       | [ROOT, Kauf]          | SHIFT     |
| [nach, München]                | [ROOT, Kauf, Tickets] |           |

Jetzt wieder SHIFT, da mit RIGHTARC „Tickets“ zu früh vom Stack genommen werden würde. Das Parsen von „nach München“ wäre dann nicht mehr möglich.

Kauf Tickets nach München.

# Shift-Reduce Dependency Parsing

| Buffer                         | Stack                 | Operation    |
|--------------------------------|-----------------------|--------------|
| [Kauf, Tickets, nach, München] | [ROOT]                | SHIFT        |
| [Tickets, nach, München]       | [ROOT, Kauf]          | SHIFT        |
| [nach, München]                | [ROOT, Kauf, Tickets] | <b>SHIFT</b> |

Kauf Tickets nach München.

# Shift-Reduce Dependency Parsing

| Buffer                         | Stack                       | Operation    |
|--------------------------------|-----------------------------|--------------|
| [Kauf, Tickets, nach, München] | [ROOT]                      | SHIFT        |
| [Tickets, nach, München]       | [ROOT, Kauf]                | SHIFT        |
| [nach, München]                | [ROOT, Kauf, Tickets]       | SHIFT        |
| [München]                      | [ROOT, Kauf, Tickets, nach] | <b>SHIFT</b> |

Kauf Tickets nach München.

# Shift-Reduce Dependency Parsing

| Buffer                         | Stack                                | Operation |
|--------------------------------|--------------------------------------|-----------|
| [Kauf, Tickets, nach, München] | [ROOT]                               | SHIFT     |
| [Tickets, nach, München]       | [ROOT, Kauf]                         | SHIFT     |
| [nach, München]                | [ROOT, Kauf, Tickets]                | SHIFT     |
| [ ]                            | [ROOT, Kauf, Tickets, nach, München] |           |

Welche Operation führen wir als nächstes durch?

Kauf Tickets nach München.

# Shift-Reduce Dependency Parsing

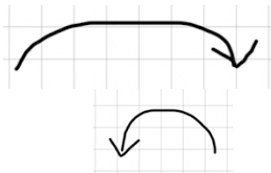
| Buffer                         | Stack                                | Operation |
|--------------------------------|--------------------------------------|-----------|
| [Kauf, Tickets, nach, München] | [ROOT]                               | SHIFT     |
| [Tickets, nach, München]       | [ROOT, Kauf]                         | SHIFT     |
| [nach, München]                | [ROOT, Kauf, Tickets]                | SHIFT     |
| [ ]                            | [ROOT, Kauf, Tickets, nach, München] | LEFTARC   |



Kauf Tickets nach München.

# Shift-Reduce Dependency Parsing

| Buffer                         | Stack                                | Operation |
|--------------------------------|--------------------------------------|-----------|
| [Kauf, Tickets, nach, München] | [ROOT]                               | SHIFT     |
| [Tickets, nach, München]       | [ROOT, Kauf]                         | SHIFT     |
| [nach, München]                | [ROOT, Kauf, Tickets]                | SHIFT     |
| [ ]                            | [ROOT, Kauf, Tickets, nach, München] | LEFTARC   |
| [ ]                            | [ROOT, Kauf, Tickets, München]       | RIGHTARC  |

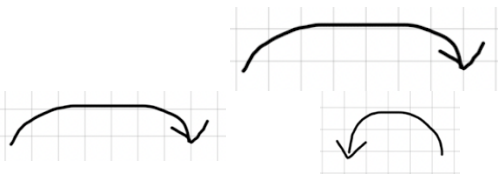


Kauf Tickets nach München.



# Shift-Reduce Dependency Parsing

| Buffer                         | Stack                                | Operation       |
|--------------------------------|--------------------------------------|-----------------|
| [Kauf, Tickets, nach, München] | [ROOT]                               | SHIFT           |
| [Tickets, nach, München]       | [ROOT, Kauf]                         | SHIFT           |
| [nach, München]                | [ROOT, Kauf, Tickets]                | SHIFT           |
| [ ]                            | [ROOT, Kauf, Tickets, nach, München] | LEFTARC         |
| [ ]                            | [ROOT, Kauf, Tickets, München]       | RIGHTARC        |
| [ ]                            | [ROOT, Kauf, Tickets]                | <b>RIGHTARC</b> |



Kauf Tickets nach München.

# Shift-Reduce Dependency Parsing

| Buffer                         | Stack                                | Operation |
|--------------------------------|--------------------------------------|-----------|
| [Kauf, Tickets, nach, München] | [ROOT]                               | SHIFT     |
| [Tickets, nach, München]       | [ROOT, Kauf]                         | SHIFT     |
| [nach, München]                | [ROOT, Kauf, Tickets]                | SHIFT     |
| [ ]                            | [ROOT, Kauf, Tickets, nach, München] | LEFTARC   |
| [ ]                            | [ROOT, Kauf, Tickets, München]       | RIGHTARC  |
| [ ]                            | [ROOT, Kauf, Tickets]                | RIGHTARC  |
| [ ]                            | [ROOT, Kauf]                         | RIGHTARC  |

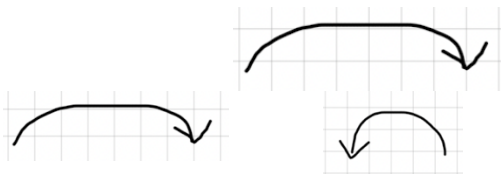


Kauf Tickets nach München.

ROOT

# Shift-Reduce Dependency Parsing

| Buffer                        | Stack                                | Operation   |
|-------------------------------|--------------------------------------|-------------|
| [Kauf Tickets, nach, München] | [ROOT]                               | SHIFT       |
| [Tickets, nach, München]      | [ROOT, Kauf]                         | SHIFT       |
| [nach, München]               | [ROOT, Kauf, Tickets]                | SHIFT       |
| [ ]                           | [ROOT, Kauf, Tickets, nach, München] | LEFTARC     |
| [ ]                           | [ROOT, Kauf, Tickets, München]       | RIGHTARC    |
| [ ]                           | [ROOT, Kauf, Tickets]                | RIGHTARC    |
| [ ]                           | [ROOT, Kauf]                         | RIGHTARC    |
| [ ]                           | [ROOT]                               | <b>DONE</b> |



Kauf Tickets nach München.

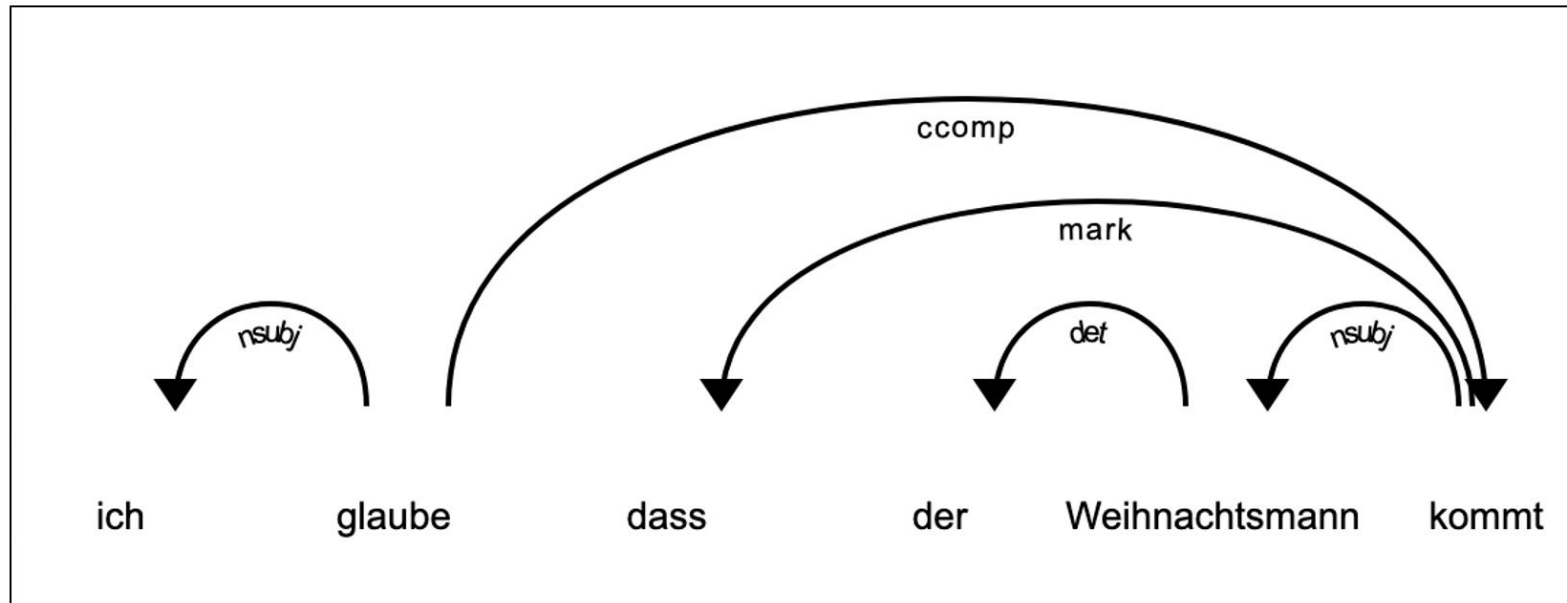
**ROOT**

# Shift-Reduce Dependency Parsing

## Projektive Struktur:

Für jede Dependenzrelation im Satz gilt, dass es einen Pfad vom Kopf zu allen Wörtern zwischen Kopf und Dependent gibt.

Beispiel:



Z.B. können wir jedes Wort zwischen „glaube“ (Kopf) und „kommt“ (Dependent) von „glaube“ aus erreichen.

# Shift-Reduce Dependency Parsing

## Nicht-projektive Struktur:

Es gibt Dependenzrelation im Satz, für die die Bedingung für projektive Strukturen nicht gilt, d.h. wir können **nicht** bei allen Dependenzrelationen im Satz jedes Wort zwischen Kopf und Dependent erreichen.

Beispiel:



In diesem Beispiel ist “talk“ Kopf von “about“, und „yesterday“ steht zwischen Kopf und Dependent.

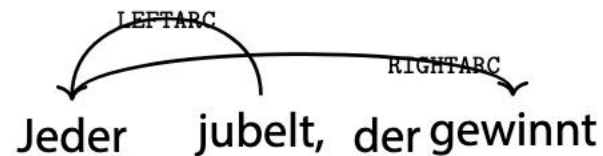
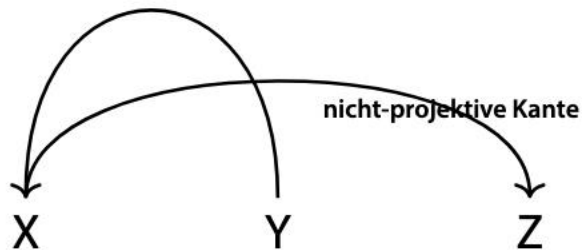
Es gibt keinen Pfad vom Kopf („talk“) zu “yesterday“, daher handelt es sich um eine nicht-projektive Struktur.

vgl. Vorlesungsfolien

# Shift-Reduce Dependency Parsing

Beim Dependency-Parsing können nicht-projektive Strukturen problematisch sein.

Der übergangsbasierte **Shift-Reduce-Dependency-Parser kann beispielsweise mit nicht-projektiven Strukturen nicht umgehen.**

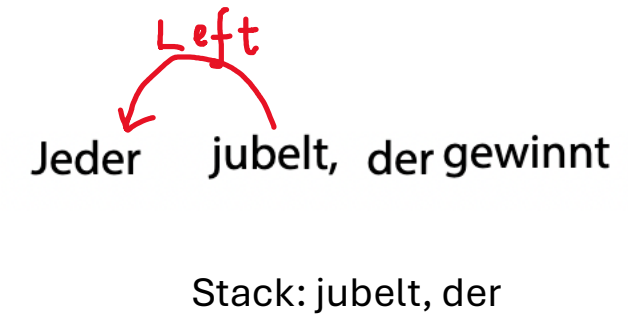
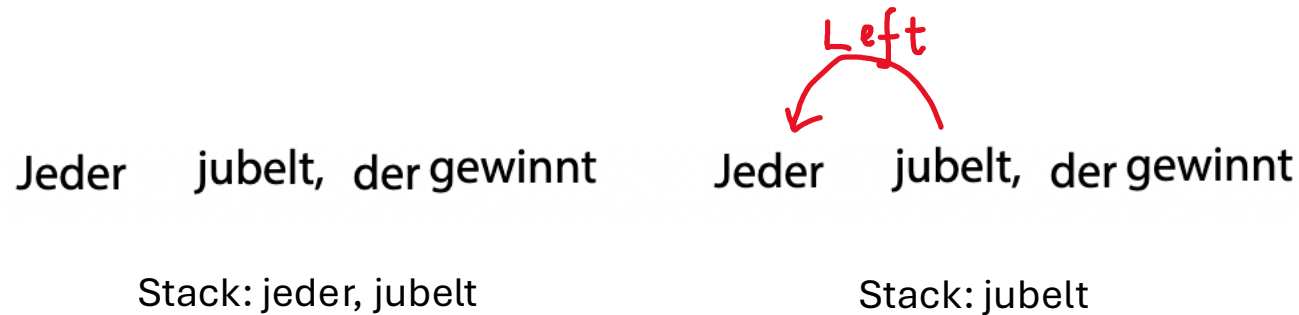


vgl. Vorlesungsfolien

## Was ist das Problem bei Parsen einer solchen Struktur?

Wir würden hier X und Y auf das Stack shiften, dann REDUCE-LEFTARC anwenden und den Dependenten (X) vom Stack nehmen. Wenn wir X vom Stack nehmen, können wir aber den Rest des Satzes nicht mehr parsen!

# Shift-Reduce Dependency Parsing



„jeder“ ist schon weg. Kann nicht mehr geparkt werden!