

Einführung in die Computerlinguistik

Syntax II

WS 2008/2009

Manfred Pinkal

Eigenschaften der syntaktischen Struktur [1]

- *Er hat die Übungen gemacht.*
- *Der Student hat die Übungen gemacht.*
- *Der interessierte Student hat die Übungen gemacht.*
- *Der an computerlinguistischen Fragestellungen interessierte Student hat die Übungen gemacht.*
- *Der an computerlinguistischen Fragestellungen interessierte Student im ersten Semester hat die Übungen gemacht.*
- *Der an computerlinguistischen Fragestellungen interessierte Student im ersten Semester, der im Hauptfach Informatik studiert, hat die Übungen gemacht.*
- *Der an computerlinguistischen Fragestellungen interessierte Student im ersten Semester, der im Hauptfach, für das er sich nach langer Überlegung entschieden hat, Informatik studiert, hat die Übungen gemacht.*

Morphologie und Syntax

- Gegenstand der **Morphologie** ist die **Struktur des Wortes**: der Aufbau von Wörtern aus Morphemen, den kleinsten funktionalen oder bedeutungstragenden Einheiten der Sprache.
- Gegenstand der **Syntax** ist die **Struktur des Satzes**: der Aufbau von Sätzen aus Wörtern.
- **Morphologie** beschreibt die **grammatischen Eigenschaften von Wörtern**, die durch Wortform oder Flexionsmorpheme kodiert werden.
- **Syntax** beschreibt die **Interaktion der grammatischen Eigenschaften** unterschiedlicher Wörter im Satz.

Eigenschaften der syntaktischen Struktur [1]

- Sätze setzen sich aus Bestandteilen zusammen, die wiederum aus einfachen oder ihrerseits komplexen Satzteilen bestehen können. Sätze können deshalb **beliebig lang und beliebig tief geschachtelt** sein.

Eigenschaften der syntaktischen Struktur [2]

Peter hat der Dozentin das Übungsblatt heute ins Büro gebracht.

Das Übungsblatt hat Peter der Dozentin heute ins Büro gebracht.

Der Dozentin hat Peter heute das Übungsblatt ins Büro gebracht.

Ins Büro hat heute Peter der Dozentin das Übungsblatt gebracht.

Heute hat Peter das Übungsblatt der Dozentin ins Büro gebracht.

Ins Büro hat das Übungsblatt der Dozentin Peter heute gebracht.

** Ins Büro heute Peter das Übungsblatt hat gebracht der Dozentin.*

** Ins heute Büro der Peter Dozentin das hat Übungsblatt gebracht.*

Eigenschaften der syntaktischen Struktur

- Sätze setzen sich aus Satzteilen (Konstituenten) zusammen, die wiederum aus einfachen oder ihrerseits komplexen Satzteilen bestehen können. Sätze können deshalb beliebig lang und beliebig tief geschachtelt sein.
- Die Syntax natürlicher Sprachen erlaubt **variable Wortstellung**: Wörter und Ausdrücke mit der gleichen Funktion können an unterschiedlichen Stellen eines Satzes stehen. Unterschiedliche Sprachen erlauben sehr unterschiedliche Freiheitsgrade.

Eigenschaften der syntaktischen Struktur [3]

- Wie finden Sie stattdessen **die** angehängten Bilder? Das **sind** Fotos, **die** im Rahmen des TALK-Projektes entstanden **sind**, uns **gehören**, und von BMW schon freigegeben **waren**. Außerdem **vermitteln sie** besser den Bezug zur Forschung.

Eigenschaften der syntaktischen Struktur [3]

- Sätze setzen sich aus Satzteilen (Konstituenten) zusammen, die wiederum aus einfachen oder ihrerseits komplexen Satzteilen bestehen können. Sätze können deshalb beliebig lang und beliebig tief geschachtelt sein.
- Die Syntax natürlicher Sprachen erlaubt variable Wortstellung: Wörter und Ausdrücke mit der gleichen Funktion können an unterschiedlichen Stellen eines Satzes stehen. Unterschiedliche Sprachen erlauben sehr unterschiedliche Freiheitsgrade.
- Die **grammatischen Eigenschaften** unterschiedlicher Ausdrücke im Satz **hängen voneinander ab** – zum Teil auch in Fällen, in denen die Ausdrücke im Satz weit auseinander liegen.

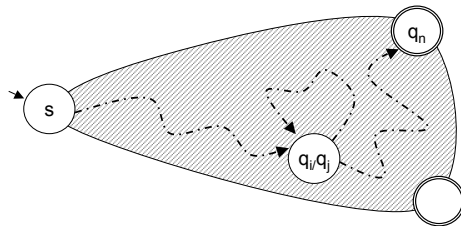
Syntax natürlicher Sprachen und endliche Automaten

- Natürliche Sprachen sind Sprachen im Sinne unserer Definitionen zur Automatentheorie:
 - Das "Alphabet" Σ ist das (eventuell sehr große) Lexikon
 - "Worte" über dem Alphabet sind Folgen von Wörtern aus dem Lexikon
 - Die deutsche Sprache lässt sich formal charakterisieren als die Sprache/ Wortmenge L über dem Alphabet (also der Folgen von Wörtern im Lexikon), die grammatisch korrekte Sätze des Deutschen darstellen.
- Frage: Lassen sich natürliche Sprachen in diesem Sinn durch endliche Automaten beschreiben? Gibt es einen NEA/DEA A , sodass $L(A)$ genau die grammatisch korrekten Sätze der deutschen Sprache enthält?
- Sprachen, die von einem endlichen Automaten erkannt werden, heißen auch "reguläre Sprachen". Die Frage ist also, kurz gefasst: **Sind natürliche Sprachen regulär?**

Sind natürliche Sprachen regulär?

- Wir zeigen, dass Sprachen, die von einem endlichen Automaten akzeptiert werden, eine bestimmte Eigenschaft haben müssen (das sog. Pumping Lemma, dt. „Pump-Lemma“ oder auch „uvw-Theorem“).
- Wir zeigen für eine bestimmte Sprache, dass sie diese Eigenschaft nicht besitzt: Es gibt nicht-reguläre Sprachen, und sie lassen sich einfach beschreiben.
- Wir argumentieren, dass natürliche Sprachen wie das Deutsche sich ebenso verhalten.

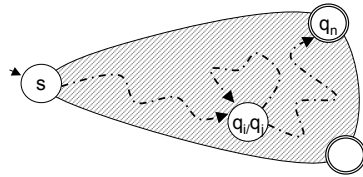
Pumping Lemma: Grundgedanke



Pumping Lemma: Grundgedanke

- Der Grundgedanke ist einfach:
 - Jede reguläre Sprache L wird von einem endlichen Automaten akzeptiert. Ein endlicher Automat hat eine bestimmte, endliche Anzahl von Zuständen.
 - Endliche Sprachen sind immer regulär. Unendliche Sprachen enthalten Worte beliebiger Länge.
 - Wenn ein Wort in L mehr Symbole hat als der Automat Zustände (genau genommen reichen schon mindestens soviele Symbole aus), dann muss beim Abarbeiten des Wortes ein Zustand mindestens zweimal vorkommen.
 - Das bedeutet, dass beim Abarbeiten eine Schleife durchlaufen wird.
 - Die kann aber auch mehrfach durchlaufen oder weggelassen werden.
 - Das heißt, dass Wörter oberhalb einer bestimmten Länge durch beliebige Wiederholung eines Teilworts „aufgepumpt“ (bzw. gekürzt) werden können und immer noch zur Sprache gehören.

Pumping Lemma: Präzise Formulierung



- Sei $w = a_1 \dots a_n$ ein Wort, das vom DEA K mit k Zuständen erkannt wird, und $n = |w| \geq k$. Dann geht der Pfad vom Startzustand $q_0 = s$ zu einem Endzustand q_n , auf dem w gelesen wird, durch insgesamt $n+1$ Zustände. Das heißt, dass mindestens zwei Zustände q_i und q_j identisch sein müssen.
- Wenn L regulär ist/ durch einen endlichen Automaten definiert wird, dann gilt:
Wenn ein Wort $x \in L$ eine bestimmte Länge k erreicht oder überschreitet ($|x| \geq k$), dann läßt sich x so in drei Teile u , v und w zerlegen (mit $|v| \geq 1$), daß mit $x = uvw$ auch jedes $x' = uv^i w$ ($i=0$ oder $i>1$) Element von L ist.
- Um zu zeigen, dass eine Sprache L nicht regulär ist, genügt es, zu zeigen, dass L ausreichend lange Worte enthält, deren Teile nicht beliebig iterierbar sind.

Vorlesung "Einführung in die CL" 2008/2009 © M. Pinkal UdS Computerlinguistik 13

Eine nicht reguläre Sprache [1]

Die Sprache $L = \{ a^n b^n \mid n \in \mathbb{N} \}$ (kurz: " $a^n b^n$ ") ist nicht regulär.

Beweis:

- Angenommen, die Sprache $L = \{ a^n b^n \mid n \in \mathbb{N} \}$ (kurz: " $a^n b^n$ ") wird von einem endlichen Automaten akzeptiert. Nach dem Pumping Lemma gibt es dann eine Zahl k , so daß für jedes Wort x mit $|x| \geq k$ eine Zerlegung in u , v und w möglich ist, so daß uw , $uvvw$, uv^2w , ... ebenfalls in L sind.
- Betrachten wir das Wort $a^k b^k$. Es gilt $|a^k b^k| \geq k$, das Wort muss also einen "duplizierbaren" Teil v besitzen. Um welchen Teil könnte es sich handeln?

Vorlesung "Einführung in die CL" 2008/2009 © M. Pinkal UdS Computerlinguistik 14

Eine nicht reguläre Sprache [2]

- Drei Fälle sind denkbar:
 - Fall1: v liegt vollständig in der ersten Hälfte des Wortes, besteht also nur aus a 's. Dann müsste gelten, dass $uv^2w = a^{k+|v|}b^k \in L$: wegen $k+|v| \neq k$ unmöglich.
 - Fall2: v liegt vollständig in der zweiten Hälfte des Wortes, besteht also nur aus b 's. Dann müsste gelten, dass $uv^2w = a^k b^{k+|v|} \in L$: wegen $k+|v| \neq k$ unmöglich.
 - Fall 3: v überspannt die Mitte des Wortes, hat also die Form $a^m b^m$. Dann müsste gelten, dass $uv^2w = a^k b^m a^m b^k \in L$. Geht nicht, da a 's auf b 's folgen.
- Es gibt also für $a^k b^k$ keine Zerlegung in uvw mit duplizierbarem Mittelteil. Also ist $L = a^n b^n$ nicht regulär.

Vorlesung "Einführung in die CL" 2008/2009 © M. Pinkal UdS Computerlinguistik 15

Was steckt hinter dem Pumping Lemma?

Endliche Automaten haben eine fundamentale Einschränkung: Ihr „Gedächtnis“ ist endlich, durch die Anzahl ihrer Zustände beschränkt. Ein Automat mit k Zuständen kann sich nur an einen beschränkten Kontext „erinnern“, nämlich maximal die k voraufgegangenen Symbole. (Anders ausgedrückt: Er kann nur bis k zählen.)

Ein endlicher Automat kann deshalb nur solche Sprachen erkennen, bei denen die Zulässigkeit eines Symbols in einer Zeichenfolge auf der Grundlage eines Vorkontextes von begrenzter Länge entschieden werden kann. Diese Eigenschaft von Sprachen heißt die „Markov-Eigenschaft“.

Um Zugehörigkeit zu $a^n b^n$ zu erkennen, müsste sich der Automat beliebig lange Ketten von a 's merken können, weil er die Information anschließend beim Abarbeiten von b 's braucht.

Vorlesung "Einführung in die CL" 2008/2009 © M. Pinkal UdS Computerlinguistik 16

Kontextfreie Grammatik: Ein neuer Formalismus

- Kontextfreie Grammatiken („KFG“, „CFG“) beschreiben Sprachen mithilfe von Ersetzungsregeln („rewrite rules“, Produktionen) der Form $A \rightarrow w$
 - Beispiel: $S \rightarrow aSb$, $S \rightarrow \epsilon$ beschreibt $L = a^n b^n$
- $A \rightarrow u$ ist zu lesen als: Ein Vorkommen von A in einer Symbolfolge/ einem Wort kann durch u ersetzt werden
 - Beispiel: $aaSbb$ wird zu $aaaSbbb$ oder zu $aa\epsilon bb = aabb$
- Eine solche Ersetzung ist ein zulässiger Ableitungsschritt. Wir schreiben: $aaSbb \Rightarrow aaaSbbb$ bzw. $aaSbb \Rightarrow aabb$.
- Um ein Wort über der Sprache $\{a, b\}$ abzuleiten, beginnen wir mit S (dem „Startsymbol“).
- Wir wenden Ersetzungsregeln an, bis ein Wort w entsteht, das nur noch a 's und b 's enthält („Terminalsymbole“).
 - Beispiel: $S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aaaSbbb \Rightarrow aaabbb$
- Wir haben damit gezeigt, dass w durch die Regeln der Grammatik aus S ableitbar ist.
- Damit ist w ein Wort der durch die Grammatik beschriebenen (erzeugten) Sprache L .

Kontextfreie Grammatik: Definitionen

$G = \langle V, \Sigma, P, S \rangle$, wobei

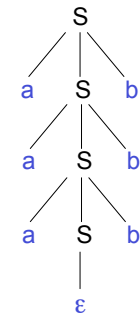
- V nicht-leere Menge von Symbolen
- $\Sigma \subseteq V$ nicht-leere Menge von **Terminalsymbolen**
- $P \subseteq (V - \Sigma) \times V^*$ nicht-leere Menge von **Produktionsregeln**
 $S \in V - \Sigma$ das **Startsymbol**

Die Beispielgrammatik für $L = a^n b^n$ in formaler Notation:

- $G_1 = \langle \{a, b, S\}, \{a, b\}, \{ \langle S, aSb \rangle, \langle S, \epsilon \rangle \}, S \rangle$
- Für $\langle A, \alpha \rangle \in P$ schreibt man üblicherweise $A \rightarrow \alpha$.

Kontextfreie Grammatiken

- Die Ableitung
 $S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aaaSbbb \Rightarrow aaabbb$
 kann alternativ durch den **Ableitungsbaum** rechts dargestellt werden.



- Die **Wurzel** des Baumes ist das Startsymbol. Die **Blätter** des Baums ergeben, von links nach rechts gelesen und aneinandergelagert, das abgeleitete Wort.
- Alternative Schreibweise:
 $[_s a[_s a[_s a[_s \epsilon] b] b] b]$

Kontextfreie Grammatik: Definitionen

- Wenn $A \rightarrow \alpha$ Produktion, $w = uAv$ und $w' = u\alpha v$, so ist w' aus w in einem Schritt ableitbar: $w \Rightarrow w'$
- w' ist aus w ableitbar: $w \Rightarrow^* w'$ gdw. es eine Folge von Ableitungsschritten gibt, die mit w beginnt und mit w' endet.
- Die durch G erzeugte Sprache $L(G)$ ist die Menge aller Worte über Σ^* , die aus S ableitbar sind: $L(G) = \{w \in \Sigma^* \mid S \Rightarrow^* w\}$
- Sprachen, die durch kontextfreie Grammatiken erzeugt werden, heißen **kontextfreie Sprachen**.

Kontextfreie Sprachen und reguläre Sprachen

- Kontextfreie Sprachen sind eine echte Obermenge der regulären Sprachen: Jede reguläre Sprache kann von einer CFG erzeugt werden, und es gibt kontextfreie Sprachen, die nicht regulär sind.
- Endliche Automaten verwenden **Iteration**: Der Automat läuft beliebig oft durch Schleifen und arbeitet dabei Wiederholungen gleicher Symbolfolgen ab.
- Kontextfreie Grammatiken verwenden **Rekursion**. Produktionsregeln verwenden in der Definition eines Ausdruckstyps den Ausdruckstyp selbst: Nicht-Terminale Symbole tauchen auf der linken und der rechten Seite von Regeln auf. Die Regel $S \rightarrow aSb$ besagt, dass ein Ausdruck, der mit einem a beginnt, mit einem b endet und dazwischen einen korrekten Ausdruck des Typs S enthält, ebenfalls ein korrekter Ausdruck vom Typ S ist.
- Rekursive Regeln erlauben die tiefe Schachtelung von Strukturen, und sie ermöglichen, dass eine Regel Elemente in Beziehung setzt, die in der Kette beliebig weit voneinander entfernt sind.

Eigenschaften der syntaktischen Struktur [1]

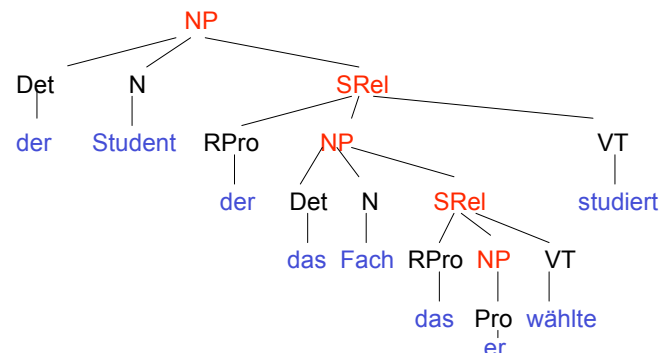
- *Er hat die Übungen gemacht.*
- *Der Student hat die Übungen gemacht.*
- *Der interessierte Student hat die Übungen gemacht.*
- *Der an computerlinguistischen Fragestellungen interessierte Student hat die Übungen gemacht.*
- *Der an computerlinguistischen Fragestellungen interessierte Student im ersten Semester hat die Übungen gemacht.*
- *Der an computerlinguistischen Fragestellungen interessierte Student im ersten Semester, der im Hauptfach Informatik studiert, hat die Übungen gemacht.*
- *Der an computerlinguistischen Fragestellungen interessierte Student im ersten Semester, der im Hauptfach, für das er sich nach langer Überlegung entschieden hat, Informatik studiert, hat die Übungen gemacht.*

Kontextfreie Sprachen und natürliche Sprachen

- Kontextfreie Grammatiken bilden den Standard-Formalismus zur syntaktischen Beschreibung von **formalen Sprachen** (Logik, Arithmetik, Programmiersprachen).
- Ein alternatives, der CGF ähnliches Format zur Beschreibung kontextfreier Sprachen ist **BNF** (die „Backus-Naur-Form“).
- Sind kontextfreie Grammatiken ein interessanter Formalismus für die syntaktische Beschreibung **natürlicher Sprachen**?

Geschachtelte Strukturen in natürlicher Sprache

$[_{NP}$ *der an computerlinguistischen Fragestellungen interessierte Student im ersten Semester, [$_{SRel}$ *der [$_{NP}$ *das Fach, [$_{SRel}$ *das [$_{NP}$ *er*] nach langer Überlegung gewählt hat]], eifrig studiert]*]]]***



Eine kontextfreie Grammatik für deutsche Sätze

$G = \langle V, \Sigma, P, S \rangle$ mit

$V = \{S, SRel, NP, VI, VT, N, Det, RPro\} \cup \Sigma$

$\Sigma = \{schläft, arbeitet, studiert, wählte, Student, Fach, der, das, er\}$

$P =$

$S \rightarrow NP VI$	$NP \rightarrow Det N$
$S \rightarrow NP VT NP$	$NP \rightarrow Det N SRel$
$SRel \rightarrow RPro NP VT$	$NP \rightarrow Pro$
$SRel \rightarrow RPro VI$	

$VI \rightarrow schläft$	$N \rightarrow Student$
$VI \rightarrow arbeitet$	$N \rightarrow Fach$
$VT \rightarrow studiert$	$RPro \rightarrow der$
$VT \rightarrow wählte$	$RPro \rightarrow das$
$Det \rightarrow der$	$Det \rightarrow das$
$Pro \rightarrow er$	$Pro \rightarrow sie$

Konstituentenstruktur

- Eine kontextfreie Grammatik enthält nicht nur die Information, ob der Satz zur Sprache gehört, also "grammatisch ist".
- Sie ordnet außerdem den Sätzen der Sprache Ableitungsbäume zu.
- Ein Ableitungsbaum kann als Repräsentation der syntaktischen Struktur eines Satzes, der "Konstituentenstruktur" interpretiert werden.
- **Konstituenten** sind die Ausdrücke, die von einem Nicht-Terminalsymbol, einer „grammatischen Kategorie“, dominiert werden.
 - Beispiele:
 - er* ist gemäß der obigen Grammatik ein Ausdruck der Kategorie Pro
 - er - der Student - der Student, der Informatik studiert* sind Ausdrücke der Kategorie NP
 - der Informatik studiert - der arbeitet* sind Ausdrücke der Kategorie Srel
- Nicht-Terminalsymbole, die Terminale / lexikalische Ausdrücke dominieren, nennt man **Präterminale** oder **lexikalische Kategorien** (in anderem Zusammenhang auch Wortart-Tags / POS-Tags)

Eine kontextfreie Grammatik für deutsche Sätze

Notationskonventionen:

- Alternative Elemente werden durch „|“ zusammengefasst (manchmal auch durch geschweifte Klammern)
- Optionale Elemente werden durch runde Klammern notiert.

Kompaktere Notation der Grammatik:

$S \rightarrow NP VI$	$S \rightarrow NP VT NP$
$SRel \rightarrow RPro VI$	$SRel \rightarrow RPro NP VT$
$NP \rightarrow Det N (SRel)$	$NP \rightarrow Pro$
$VI \rightarrow schläft arbeitet$	$VT \rightarrow wählte studiert$
$N \rightarrow Student Fach$	$RPro \rightarrow der das$
$Det \rightarrow der das$	$Pro \rightarrow er sie$

Eigenschaften der syntaktischen Struktur [2]

Peter hat der Dozentin das Übungsblatt heute ins Büro gebracht.
Das Übungsblatt hat Peter der Dozentin heute ins Büro gebracht.
Der Dozentin hat Peter heute das Übungsblatt ins Büro gebracht.
Ins Büro hat heute Peter der Dozentin das Übungsblatt gebracht.
Heute hat Peter das Übungsblatt der Dozentin ins Büro gebracht.
Ins Büro hat das Übungsblatt der Dozentin Peter heute gebracht.
** Ins Büro heute Peter das Übungsblatt hat gebracht der Dozentin.*
** Ins heute Büro der Peter Dozentin das hat Übungsblatt gebracht.*

Konstituentenstruktur

- Die Zerlegung von Sätzen in Konstituenten und ihre Kategorisierung ist nicht beliebig: Konstituenten und Kategorien müssen so gewählt werden, dass sich mit ihnen syntaktische Regularitäten /Generalisierungen beschreiben lassen.
- Beispiele:
 - Im deutschen Hauptsatz steht das Verb an zweiter Stelle: allerdings nicht in Wörtern gezählt, sondern in „Phrasen“. Diese und ähnliche Beobachtungen bilden ein Argument für die Annahme von „Phrasalen Kategorien“ als einer Grundebene der syntaktischen Beschreibung.
 - Nominalausdrücke kommen in vielen Positionen im Satz vor (Subjekt, Objekt, Genitivattribut) und sind (modulo Flexionsform) füreinander einsetzbar. Dies ist eine Motivation für die Kategorie „Nominalphrase“.
 - Eine zweite Motivation: Nominalausdrücke haben ähnliche interne Struktur: Sie besitzen z.B. als „Kopf“ ein Substantiv oder ein Pronomen.
- In der Syntax natürlicher Sprachen spricht man auch von **Konstituentenstruktur-Grammatiken** („KSG“) und von Konstituentenstruktur-Regeln.
- Die CFG oder KSG als solche ist ein **Formalismus** zur syntaktischen Beschreibung. Wenn Kategorieninventare und Regeltypen für die angemessene Beschreibung natürlicher Sprachen im Rahmen des Formalismus spezifiziert werden, entstehen **Grammatiktheorien**.

Konstituentenstruktur und Semantik

- Die syntaktische Struktur ist eine Grundlage für die semantische Interpretation. Die syntaktische Analyse muss deshalb die Interpretation unterstützen.
- Ein semantisches Argument für die Annahme der Kategorie NP: Mit Nominalausdrücken referieren wir auf Gegenstände.
- Grammatische Regeln erlauben die Modellierung struktureller Mehrdeutigkeit.

Phrasale Kategorien

- Entsprechend den **lexikalischen Hauptkategorien** Substantiv, Adjektiv, Verb und Präposition geht man üblicherweise von vier **phrasalen Hauptkategorien** aus:
 - Nominalphrasen:** *er – der Student – der interessierte Student – die Übungen – computerlinguistischen Fragestellungen*
 - Präpositionalphrasen:** *an computerlinguistischen Fragestellungen – im ersten Semester, – nach langer Überlegung*
 - Adjektivphrasen:** *interessierte – an computerlinguistischen Fragestellungen interessierte*
 - Verbphrasen:** *arbeitet - studiert Informatik - entscheidet sich für das Fach*
- Achtung: Unterschiedliche Grammatiktheorien benutzen ein zum Teil sehr unterschiedliches Kategorien-Inventar. Allerdings ist die Unterteilung in lexikalische, phrasale und „andere“ Kategorien allgemein.

Grammatische Mehrdeutigkeit, Beispiel

- Ein strukturell mehrdeutiger Satz:

Peter sah den Mann mit dem Teleskop
- Eine Grammatik, die die Mehrdeutigkeit modelliert:

$S \rightarrow NP VP$	$VP \rightarrow VT NP$	$VP \rightarrow VI$
$NP \rightarrow ART NN$	$NP \rightarrow PN$	$NP \rightarrow NP PP$
$PP \rightarrow P NP$	$VP \rightarrow VP PP$	
- Die zwei Analysevarianten:
 - $[_S \text{ Peter } [_{VP} \text{ sah } [_{NP} [_{NP} \text{ den Mann }] [_{PP} \text{ mit dem Teleskop }]]]]]$
 - $[_S \text{ Peter } [_{VP} [_{VP} \text{ sah } [_{NP} \text{ den Mann }]]] [_{PP} \text{ mit dem Teleskop }]]]$