

Einführung in die Computerlinguistik

Syntax

WS 2006/2007

Manfred Pinkal

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]

- 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.

Eigenschaften der syntaktischen Struktur [1]

- 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.

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 Konstituenten mit der gleichen Funktion können an unterschiedlichen Stellen eines Satzes stehen. Unterschiedliche Sprachen erlauben sehr unterschiedliche Freiheitsgrade.

Eigenschaften der syntaktischen Struktur [3]

- *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.*

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 Konstituenten mit der gleichen Funktion können an unterschiedlichen Stellen eines Satzes stehen. Unterschiedliche Sprachen erlauben sehr unterschiedliche Freiheitsgrade.
- Die **grammatischen Eigenschaften** unterschiedlicher Wörter und Konstituenten im Satz **hängen voneinander ab** – zum Teil auch in Fällen, in denen die Wörter und Konstituenten im Satz weit auseinander liegen.

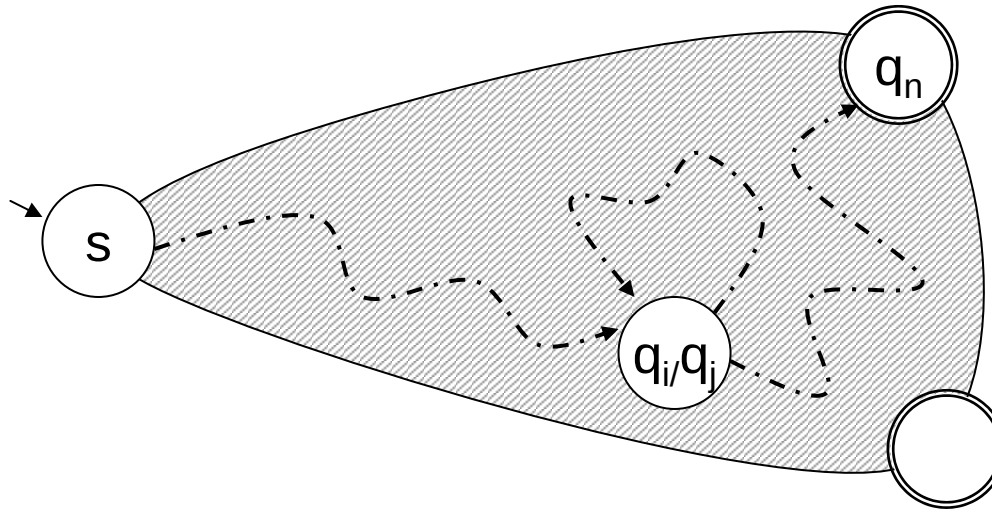
Syntax natürlicher Sprachen und endliche Automaten

- Natürliche Sprachen sind Sprachen im Sinne unserer Definitionen zur Automatentheorie:
 - Das "Alphabet" 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 Menge der Worte 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?
- Terminologie: Sprachen, die von einem endlichen Automaten erkannt werden, heißen auch "reguläre Sprachen". Die Frage ist also, kurz gefasst: Sind natürliche Sprachen regulär?

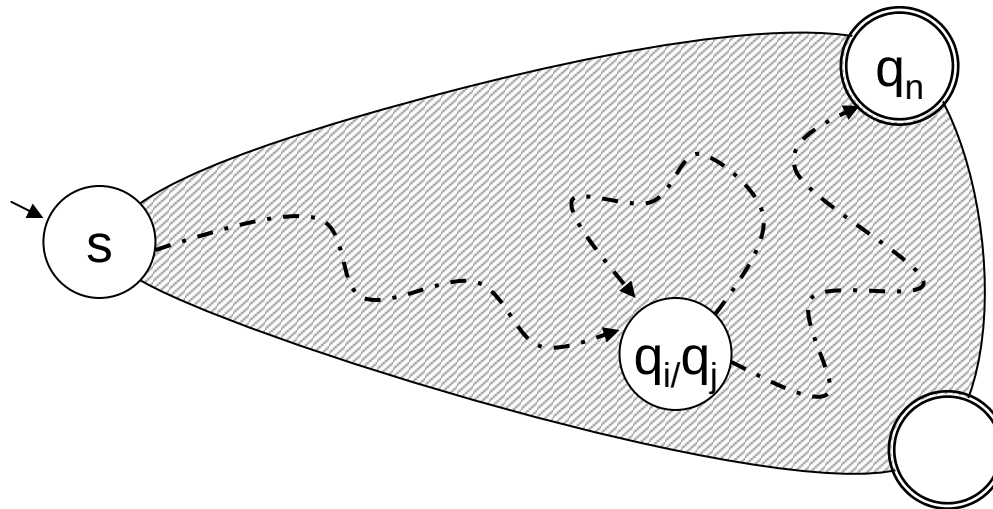
Das "Pumping Lemma,, [1]

- Das Pumping Lemma (dt. „Pump-Lemma“ oder auch „uvw-Theorem“) gibt uns, anders als das vorher gezeigte Theorem, die Möglichkeit, konstruktiv zu beweisen, dass eine Sprache nicht regulär ist.
- Der Grundgedanke ist einfach: Jede reguläre Sprache L wird von einem endlichen Automaten akzeptiert. Ein endlicher Automat hat eine bestimmte, endliche Anzahl von Zuständen. Wenn ein Wort in L mehr Symbole hat als der Automat Zustände (genau genommen reichen schon mindestens so viele Symbole aus), dann muss beim Abarbeiten des Wortes ein Zustand mindestens zweimal vorkommen. Das bedeutet aber, dass beim Abarbeiten eine Schleife durchlaufen wird. Die könnte im Prinzip aber auch mehrfach durchlaufen (oder weggelassen) werden. Das heißt, für Wörter oberhalb einer bestimmten Länge durch beliebige Wiederholung eines Teilworts „aufgepumpt“ werden können und immer noch zur Sprache gehören.

Das "Pumping Lemma"



Das "Pumping Lemma"



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.

Das "Pumping Lemma"

Präzise Formulierung des Lemmas:

- Wenn L eine Sprache ist, die durch einen endlichen Automaten beschrieben werden kann ("reguläre Sprache"), 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.

Eine nicht reguläre Sprache [1]

Die Sprache $L = \{ a^n b^n \mid n \in \mathbb{N} \}$ (kurz: " $a^n b^n$ ") wird nicht von einem endlichen Automaten akzeptiert.

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 , uvw , $uvvw$, ... 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?

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 $uv^2w = a^kb^{k+|v|} \in L$: wegen $k+|v| \neq k$ unmöglich.
 - Fall 3: v überspannt die Mitte des Wortes, hat also die Form a^lb^m . Dann müsste gelten, dass $uv^2w = a^kb^ma^lb^k \in L$. Geht nicht, da a 's auf b 's folgen.
- Es gibt also für a^kb^k keine Zerlegung in uvw mit duplizierbarem Mittelteil. Also ist $L = a^n b^n$ nicht regulär.

Was steckt hinter dem Pumping Lemma?

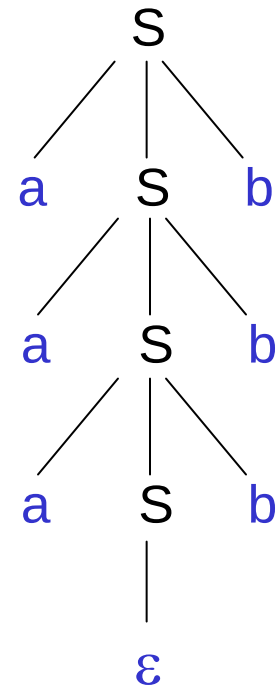
Endliche Automaten haben eine fundamentale Einschränkung: Ihr „Gedächtnis“ ist endlich, durch die Anzahl ihrer Zustände beschränkt. Ein Automat mit k Zuständen kann sich nur an einen beschränkten Kontext „erinnern“: nach spätestens k gelesenen Symbolen ist er wieder in dem Zustand, den er vorher gehabt hat. Noch anders ausgedrückt: Der Automat kann nur bis k zählen. Um zum Beispiel $a^n b^n$ zu erkennen, müsste er sich aber beliebig lange Ketten von a 's merken können, weil er die Information anschließend beim Abarbeiten von b 's braucht.

Ein Formalismus zur Beschreibung nicht-regulärer Sprachen: Kontextfreie Grammatiken

- $S \rightarrow aSb$, $S \rightarrow \varepsilon$ sind Produktionsregeln (Ersetzungsregeln, "rewrite rules")
- Beginne mit S (Startsymbol). Wenn in einer Kette das Symbol S vorkommt, ersetze es durch aSb (Regel 1) oder die leere Kette (Regel 2).
- Ersetze solange, bis nur noch a 's und b 's vorkommen („Terminalsymbole“)
- Eine gültige Ableitung der Beispielgrammatik:
 $S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aaaSbbb \Rightarrow aaabbbb$
- Die Beispielgrammatik erzeugt die Sprache $a^n b^n$

Kontextfreie Grammatiken: Ein Beispiel

- Das Wort aaabbbb wird durch
 $S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aaaSbbb \Rightarrow aaabbbb$
abgeleitet.
- Die Ableitung kann durch den folgenden **Ableitungsbaum** dargestellt werden:
- Die **Blätter** des Baums ergeben, von links nach rechts gelesen, das abgeleitete Wort.
- Alternative Schreibweise:
 $[_S a[_S a[_S a[_S \varepsilon] b] b] b]$
Kontextfreie Grammatiken ermöglichen die Darstellung beliebig tief geschachtelter Strukturen.
- „Ableitungsbaum“ heißt auch „**Parse-Baum**“ oder „**Parse-Tree**“: **Parsing** ist automatische syntaktische Analyse



Kontextfreie Grammatik: Formale Definition

$G = \langle V, \Sigma, P, S \rangle$, 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 in formaler Notation:

- $G_1 = \langle \{a, b, S\}, \{a, b\}, \{ \langle S, aSb \rangle, \langle S, \varepsilon \rangle \}, S \rangle$
- Für $\langle A, \alpha \rangle \in P$ schreibt man üblicherweise $A \rightarrow \alpha$.
- **Ableitungsschritt**: Wenn w ein nicht-terminales Symbol A enthält und $A \rightarrow \alpha$ Produktion ist, ersetze A in w durch α .
- Die durch G **erzeugte Sprache** ist die Menge aller Worte über Σ^* , die sich aus S ableiten lassen.

Kontextfreie Grammatiken und natürliche Sprachen

- Kontextfreie Grammatiken bilden den Standard-Formalismus zur syntaktischen Beschreibung von **formalen Sprachen** (Logik, Arithmetik, Programmiersprachen).
- Auch die Syntax **natürlicher Sprachen** kann einfach und elegant mithilfe von kontextfreien Grammatiken dargestellt werden.

Geschachtelte Strukturen in natürlicher Sprache

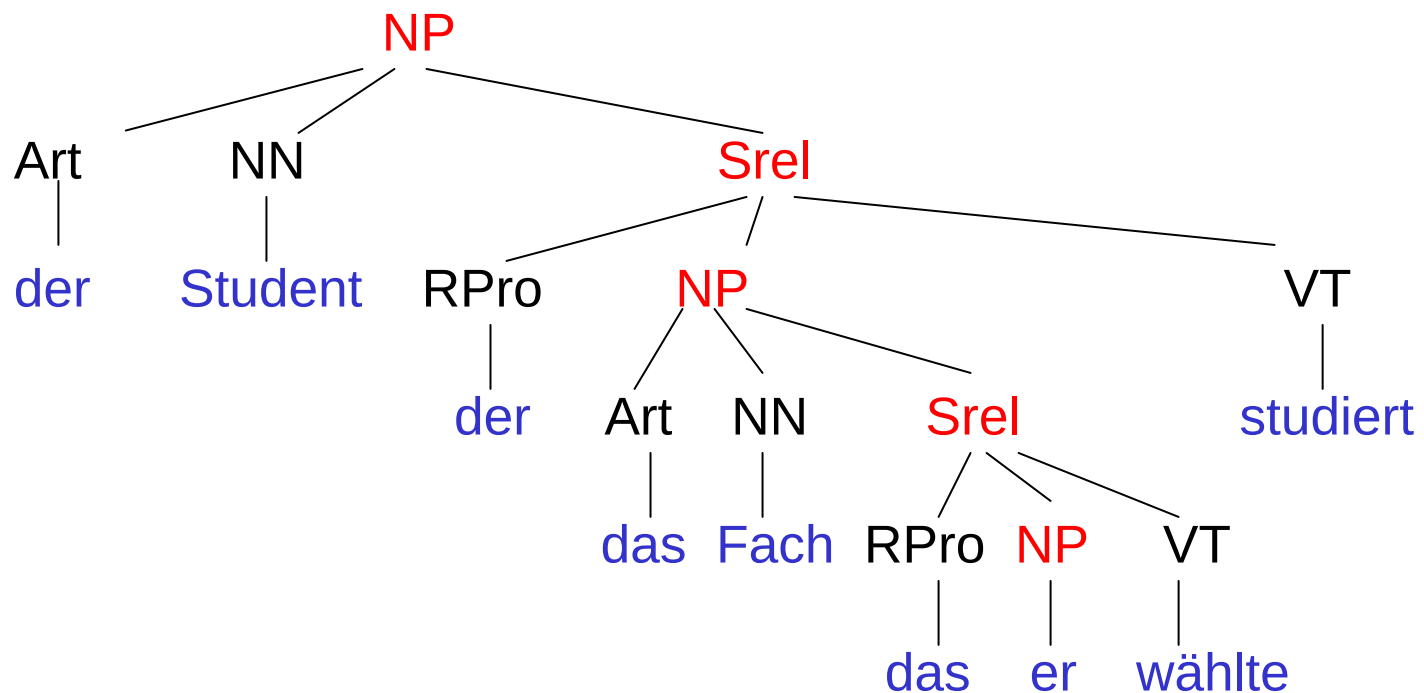
- Der **Nominalausdruck** „*der an computerlinguistischen Fragestellungen interessierte Student im ersten Semester, der im Hauptfach, für das er sich nach langer Überlegung entschieden hat, Informatik studiert*“ enthält
- den (Relativ-)**Satz** „*der im Hauptfach, für das er sich nach langer Überlegung entschieden hat, Informatik studiert*“; der enthält
- den **Nominalausdruck** „*Hauptfach, für das er sich nach langer Überlegung entschieden hat*“; der enthält
- den (Relativ-)**Satz** „*für das er sich nach langer Überlegung entschieden hat*“; der enthält
- unter anderem den **Nominalausdruck** „*er*“.

Syntaktische Kategorien

- Konstituenten-Typen werden „syntaktische Kategorien“ genannt.
Beispiele:
 - **Nominalphrasen** (Nominalausdrücke): *er – der Student – der interessierte Student – die Übungen – computerlinguistischen Fragestellungen*
 - **Präpositionalphrasen** (Präpositionalausdrücke): *an computerlinguistischen Fragestellungen – im ersten Semester, – nach langer Überlegung*
 - **Adjektivphrasen**: *interessierte – an computerlinguistischen Fragestellungen interessierte*
 - **Satz**: Haupt- und Nebensätze unterschiedlicher Art
- Sie werden in kontextfreien Grammatiken als nicht-terminale Symbole verwendet.

Geschachtelte Strukturen in natürlicher Sprache

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



Eine kontextfreie Grammatik für deutsche Sätze

$S \rightarrow NP VI$

$S \rightarrow NP VT NP$

$NP \rightarrow ART NN$

$VI \rightarrow schl\ddot{a}ft$

$VI \rightarrow arbeitet$

$VT \rightarrow studiert$

$VT \rightarrow w\ddot{a}hlt$

$ART \rightarrow der$

$NP \rightarrow ART NN SREL$

$SREL \rightarrow RPRO VI$

$SREL \rightarrow RPRO NP VT$

$NN \rightarrow Student$

$NN \rightarrow Fach$

$RPro \rightarrow der$

$RPro \rightarrow das$

$ART \rightarrow das$

- Kontextfreie Grammatiken heien in der Linguistik auch **Konstituentenstruktur-Grammatiken** oder **Phrasenstruktur-Grammatiken**

Eine kontextfreie Grammatik für deutsche Sätze

- Schreibkonventionen: Klammernotation für optionale Konstituenten, Reihung von lexikalischen Alternativen

$S \rightarrow NP\ VI$ $NP \rightarrow ART\ NN\ (SREL)$

$SREL \rightarrow RPRO\ VI$ $S \rightarrow NP\ VT\ NP$

$SREL \rightarrow RPRO\ NP\ VT$

$VI \rightarrow \textit{schläft, arbeitet, wartet, ...}$

$VT \rightarrow \textit{wählt, studiert, liest, kennt, ...}$

$NN \rightarrow \textit{Student, Fach, Dozentin, Professor, Buch, ...}$

$RPro \rightarrow \textit{der, die, das}$

$ART \rightarrow \textit{der, die, das}$

Konstituentenstruktur

- Die Erzeugung eines Satzes mit einer kontextfreien Grammatik (und die Verarbeitung des Satzes mit einem Parser, der die Grammatik in ein Analysesystem umsetzt) liefert nicht nur die Information, ob der Satz zur Sprache gehört/"grammatisch ist". Als "Seiteneffekt" liefert sie den Ableitungsbaum/"Parse Tree", der die "**Konstituentenstruktur**" und damit einen wesentlichen teil der syntaktischen Struktur beschreibt.
- Eine Grammatik ist eine **angemessene Beschreibung** der syntaktischen Struktur einer Sprache, wenn sie
 - genau die grammatisch akzeptablen Sätze der Sprache erzeugt, und außerdem
 - für die grammatischen Sätze angemessene Konstituentenstrukturen erzeugt.

Grammatische Mehrdeutigkeit

- Natürlich-sprachliche Sätze sind oft grammatisch mehrdeutig. Die grammatische Mehrdeutigkeit ist relevant. Sie führt im allgemeinen zu semantischer Mehrdeutigkeit.
- Für natürliche Sprachen benötigen wir deshalb Grammatiken, die Sätzen unterschiedliche syntaktische Strukturen zuweisen können.

Grammatische Mehrdeutigkeit, Beispiel

- Die folgende Grammatik erlaubt Präpositionalphrasen (PPs) als VP- (Verbphrasen-) und als NP-Modifikatoren

$S \rightarrow NP VP$

$VP \rightarrow VT NP$

$VP \rightarrow VI$

$NP \rightarrow ART NN$

$NP \rightarrow PN$

$NP \rightarrow NP PP$

$PP \rightarrow P NP$

$VP \rightarrow VP PP$

- Der Satz: *Peter sah den Mann mit dem Teleskop* hat in dieser Grammatik zwei Analysen, die zwei Bedeutungsvarianten entsprechen:

- $[_S \text{Peter} [_{VP} \text{sah} [_{NP} [_{NP} \text{den Mann}] [_{PP} \text{mit dem Teleskop}]]]]$
- $[_S \text{Peter} [_{VP} [_{VP} \text{sah} [_{NP} \text{den Mann}]]] [_{PP} \text{mit dem Teleskop}]]]$

- Der Satz *Peter rief den Kollegen aus München mit dem Handy an* hat mindestens 4 Lesarten (welche?).

Kategorie und Funktion

- **Syntaktische Kategorien** bezeichnen Klassen von Konstituenten, die jeweils
 - die gleichen Plätze in der syntaktischen Struktur einnehmen können (wechselseitig ersetzbar sind) und
 - Gemeinsamkeiten im internen Aufbau besitzen, insbesondere in Bezug auf die Wortart ihres „lexikalischen Kopfes“: Die Konstituenten
 - *der an computerlinguistischen Fragestellungen interessierte Student*
 - *an computerlinguistischen Fragestellungen interessiert*
 - *an computerlinguistischen Fragestellungen*sind Nominal-, Adjektiv- und Präpositionalphrase, da der jeweilige Kopf Substantiv („Nomen“), Adjektiv, bzw. Präposition ist.
- **Grammatische Funktionen** dagegen bezeichnen die Rolle, die eine Konstituente im ganzen Satz spielt. Ein Nominalausdruck z.B. kann, je nach Stellung im Satz unter anderem die Funktion von **Subjekt**, (direktem oder indirektem) **Objekt**, (Genitiv-) **Attribut** oder **Prädikatsnomen** besitzen.

Syntaktische Verarbeitung

- Kontextfreie Grammatiken erlauben die einfache und elegante Beschreibung syntaktischer Strukturen (ähnlich wie Zustandsdiagramme/deterministische endliche Automaten das für morphologische Strukturen tun).
- Wie sehen Analysesysteme für kontextfreie Sprachen aus (syntaktische **Parser**)?