

## **14 Zusammenfassungen**

**Sitzung 2: Einführung**

**Sitzung 3: Syntaktische Kategorien**

**Sitzung 4: Syntaktische Relationen: Konstituenz**

**Sitzung 5: Syntaktische Relationen: Dependenz**

**Sitzung 6: Morphologische Form syntaktischer Funktionen**

**Sitzung 7: Unifikationsgrammatiken**

**Sitzung 8: Komplexe Satzkonstruktionen und Wortstellung**

**Sitzung 9: Parsing-Algorithmen**

**Sitzung 10: Unifikation**

## **Sitzung 11: Statistische Syntaxmodelle**

11.1 Probabilistische kontextfreie Grammatiken

11.2 Statistische Abhängigkeitsmodelle

## **Sitzung 12: Datengestützte Syntaxmodelle**

12.1 Induzierte PCFG-Modelle

12.2 Parsing mit neuronalen Netzen

## **Sitzung 13: Partielles Parsing**

13.1 Partielles Parsing

13.2 Komplexität natürlicher Sprachen

# 14 Zusammenfassungen

# Sitzung 2: Einführung

- **Syntax-, Grammatik- und Satzbegriff**
- **Syntax natürlicher Sprachen**
  - Regeln der Kombination von Wörtern zu Sätzen (Satzlehre)
- **Konstituentenstruktur**
  - Analyse der Hierarchie **syntaktischer Einheiten** (Phrasenstrukturgrammatik im weiteren Sinne)
  - Strukturinformationen in Knoten des Syntaxbaums (Konstituenten = phrasale Einheiten)

- **Dependenzstruktur**

- Analyse der hierarchischen **syntaktischen Abhängigkeitsrelationen** zwischen Wörtern (Wortgrammatik)
- Strukturinformationen in Kanten des Syntaxbaums (grammatische Relationen als funktionale Kategorien)

- **formale Grammatik**

- mathematische Struktur zur Modellierung natürlichsprachlicher Satzstruktur
- kontextfreie Grammatik (CFG) als Phrasenstrukturgrammatik im engeren Sinne (PSG)

- **Parsing**

- algorithmische Verarbeitung von formalen Grammatiken zur automatischen Satzstrukturanalyse
- Erkennung der Wohlgeformtheit (Grammatikalität) einer Eingabe
- Wiedergabe der syntaktischen Struktur (Syntaxbaum)

# Sitzung 3: Syntaktische Kategorien

- Syntaktische Einheiten = Konstituenten
  - *Wörter - Phrasen - Sätze*
  - *Wörter = elementare Einheiten*
  - *Phrasen = Gruppen von Wörtern, Erweiterung um Phrasenkopf*
  - *Feststellbar durch Konstituententests*

- **Kategorisierungen syntaktischer Einheiten**

→ ***syntaktische Kategorie*** = Menge von syntaktischen Einheiten mit gleichen ***morphosyntaktischen Eigenschaften*** (Abstraktionsklasse)

→ Klassen primär definiert über ***Austauschbarkeit im gleichen Kontext***

→ ***sprachabhängig!***



- **Wortarten = Lexikalische Kategorien**

- *Hauptkategorien: **Nomen, Verb***

- *Modifikatoren: **Adjektiv, Adverb***

- *Nominale Begleiter und Proformen: **Pronomen, Determinativ***

- *Weitere Kategorien: **Adposition, Konjunktion, Partikel***

- **Phrasenkategorien**

- *definiert durch **Wortart des Phrasenkopfs***

- *nur bestimmte Wortarten sind **phrasenbildend***

- *Phrasen können **komplex** sein, d. h. andere Phrasen enthalten*

- ( $PP = P + NP$ ;  $NP = NP + PP$ )*

- ***Nominal-, Verbal-, Adjektiv-, Adverb-, Adpositional-Phrase***

# **Sitzung 4: Syntaktische Relationen: Konstituenz**

- **Konstituentenstruktur (auch: Phrasenstruktur)**

- Konstituenz = **Teil-Ganzes-Beziehung** zwischen sprachlichen Einheiten (Konstituenten)

- Relation der **unmittelbaren Dominanz** zwischen Einheit und ihren unmittelbaren Konstituenten

- **in phrasalen Einheiten** können neben lexikalischen auch **phasale Einheiten anderer oder gleicher Kategorie** vorkommen

- ⇒ **hierarchischer, rekursiver Strukturaufbau**

- **Merkmalsvererbung** vom **Kopf als Phrasenkern** an Phrase

- Köpfe werden im Syntaxbaum nach oben weitergereicht (**Perkolatio**

- Analyse **diskontinuierlicher Phrasen** über *traces* (Spuren)

- **Kontextfreie Grammatik**

- formale Grammatik mit **kontextfreien Regeln**

- verwendet zur **Modellierung der Konstituentenstruktur natürlicher Sprache**

- Phrasenstrukturgrammatik (**PSG**) im engeren Sinne

- beschreibt Regeln der **Kombination von lexikalischen und phrasalen Kategorien** (nichtterminale Symbole) zu **phrasalen Kategorien und Sätzen** (Startsymbol S)

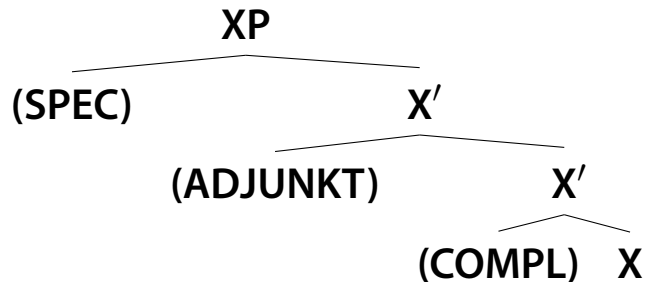
- Eine syntaktische Struktur (**Syntaxbaum**) wird von einer Grammatik erfüllt, wenn eine **Ableitung aus den als Produktionsregeln** aufgefassten Regeln der Grammatik existiert

- **X-Bar-Schema**

→ Beschränkung der Struktur: **binäre Verzweigung**:  $A \rightarrow B C$

→ Einführung phrasaler **Analyseebene zwischen Phrase und Kopf (X')**

→ gleichartiges Schema für alle Phrasen:



→ **Komplement**: Schwester von Kopf, Tochter von X'

→ **Adjunkt**: Schwester von X', Tochter von X'

→ **Spezifizierer**: Schwester von X', Tochter von XP

- **CFGs als Konstituentenstrukturmodell**

- Modellierung des **hierarchischen, rekursiven Aufbaus** natürlicher Sprache aus lexikalischen und phrasalen Kategorien
- **X-Bar: Differenzierung Argument-Adjunkt-Spezifizierer**
- Nichtberücksichtigung von Morphosyntax und Subkategorisierung → **Übergenerierung**

- **Erweiterungen von CFGs**

- Einführung **komplexerer atomarer Kategorien**
- **Merkmalsstrukturen** (Unifikationsgrammatiken)
- Auswahl durch **probabilistisches Modell** (PCFG)

# **Sitzung 5: Syntaktische Relationen: Dependenz**

- **Dependenzstruktur**

- Untersuchung der **Abhängigkeit von Vorkommen und Form** von Wörtern im Satz

- **Dependenzrelation** = binäre asymmetrische Relation zwischen Wörtern (Kopf und Dependent)

- 2 Typen von Abhängigkeiten:

- **Rektion** (*bilaterale Abhängigkeit*): → **Komplemente**

- **Modifikation** (*unilaterale Abhängigkeit*): → **Modifikatoren**

- **Valenzgrammatik**: Untersuchung ausgehend vom Verb



- **Komplement** (valenzgrammatisch: **Ergänzung / Aktant**)
  - **obligatorischer Dependent** (gefordert vom Kopf)
  - aber: kann **fakultativ** sein
- **Modifikator**
  - **optionaler Dependent**
  - hängt ab von Kopf, aber wird nicht vom Kopf gefordert
    - *verbal*: **Adjunkt** (valenzgrammatisch: **Angabe / Zirkumstant**)
    - *nominal*: **Attribut**

- **Dependenzrelationen als syntaktische Funktionen**
  - **Kategorisierung der Dependenzrelationen nach syntaktischem Verhalten der Dependenden**
  - Feststellung der **syntaktischen Funktion** einer Einheit, die sie in Bezug auf ihren Kopf einnimmt (z.B. Objekt-Komplement)
- **Grammatische Relationen → syntaktische Funktion verbaler Dependenden (= Satzglieder)**
  - **Subjekt:** Kernargument intransitiver Satz, Kongruenz mit Verb
  - **Objekt:** passivierbares Patiens-Argument transitiver Satz
  - **indirektes Objekt:** Recipient-Argument ditransitiver Satz
  - **Adverbial:** nicht-zentrales, peripheres Argument

- **Attributfunktionen → Syntaktische Funktion nominaler Modifikatoren**
  - Adjektiv-/Partizipial-Attribut
  - Präpositionales Attribut
  - Genitiv-Attribut
  - Determinativ
  - Apposition
  - Attributsatz

- **Dependenzgrammatik**

- formale Repräsentation als **gerichteter Graph**

- **Wortgrammatik**

- Strukturinformation in den Kanten (Relationen)

- Transformation Konstituenten- in Dependenzstruktur möglich

- Hauptvorteil gegenüber PSGs: **Grammatische Funktionen direkt kodiert**

- **Übersicht: Adverbial, Angabe, Ergänzung, Präpositionalobjekt**

Dependenztyp	syntaktische Funktion	Auftreten	Form	Beispiel
Komplement/Ergänzung	Subjekt / Objekt auch Präpositionalobjekt:	<b>valenzgefordert</b>	<b>valenzgefordert</b>	<i>jemandes gedenken</i> <i>an jmd. denken</i>
Komplement/Ergänzung	Adverbial	<b>valenzgefordert</b>	<i>nicht</i> valenzgefordert	<i>auf den Tisch / ins Wasser stellen</i>
Adjunkt/Angabe	Adverbial auch Kasusadverbial:	<i>nicht</i> valenzgefordert	<i>nicht</i> valenzgefordert	<i>Es regnet (im Park / auf den Tisch)</i> <i>Es geschieht <b>dieser</b> Tage</i>

– **Präpositionaladverb vs. Präpositionalobjekt:**

- \* *Er wartet **auf dem Berg*** (adverbiale Angabe: wo?, Dativ)  
***auf die Sonne*** (Präpositionalobjekt: worauf?, Akk.).
- \* **Angabe (*auf dem Berg*) optional:** *Er wartet auf die Sonne.*
- \* **Das präpositionale Komplement von *warten* ist fakultativ:** *Er wartet (*auf die Sonne*).*

– **Adverbiales Komplement:**

- \* **Verb verlangt keine Formeigenschaft!**
- \* *Er stellt die Blumen **auf den Tisch*** (adverbiales Komplement).
- \* *Er stellt die Blumen **ins Wasser*** (adverbiales Komplement).

# Sitzung 6: Morphologische Form syntaktischer Funktionen

- **Sprachliche Ausdrucksmittel syntaktischer Funktionen**
  - strukturell über **Wortstellung**
  - morphologisch über **Flexionsmorphologie**
- **morphologische Kodierung grammatischer Relationen über:**
  - **Kasus:** Markierung der Funktion der Relation zwischen Verb und Dependent durch **Marker am Dependent** (Rektion)
  - **Agreement:** Markierung der Funktion der Relation zwischen Verb und Dependent durch **Merkmalskongruenz**

- **Funktionale Kategorien**

→ Funktionale Syntax: Untersuchung der **systematischen Variation von morphosyntaktischer Kodierung** mit **semantischer und pragmatischer Rolle**

→ **Diathesen**: syntaktische Operation der **Manipulation der Abbildung semantischer Rollen** auf Grammatische Relationen

→ *Passivierung: Promotion des Patiens-Arguments in Subjektposition*

→ **Topik-Fokus-Struktur**: **kontextabhängige**, pragmatische **Struktur der Äußerung**, die u. a. über syntaktische Operationen wie **Linksversetzung** oder **Cleftsätze** angezeigt werden kann



- **Morphosyntaktische Typologien**

- **Varianz** in der Kodierung syntaktischer Funktionen im **Sprachvergleich**

- Systematische **Differenz in der Abbildung semantischer Rollen** auf Grammatische Relationen: **Akkusativ- vs. Ergativsprachen**

- **Aktiv-Sprachen** wie das Georgische kodieren primär die **semantische Rolle**

- **Topik-prominente Systeme** wie das Japanische kodieren primär die **pragmatische Rolle**

- **Merkmalsstrukturen**

- **formale Repräsentation von grammatischen Kategorien**
- **atomare oder komplexe Werte** (Merkmalsstruktur als Wert, z.B. für Bündelung von Agreementmerkmale)
- **Beschreibung** von lexikalischen Einheiten und Kategorien als komplexe Objekte, die über **Merkmale** definiert sind:

Wortformen: *Hund*  $\left[ \begin{array}{cc} \text{CAT} & N \\ \text{AGR} & \left[ \begin{array}{cc} \text{NUM} & SG \\ \text{GEN} & MASK \end{array} \right] \end{array} \right]$ , *der*  $\left[ \begin{array}{cc} \text{CAT} & DET \\ \text{AGR} & \left[ \begin{array}{cc} \text{NUM} & SG \\ \text{GEN} & MASK \\ \text{CASE} & NOM \end{array} \right] \end{array} \right]$

lexikalische Kategorien:  $\left[ \begin{array}{cc} \text{CAT} & N \end{array} \right]$   $\left[ \begin{array}{cc} \text{CAT} & DET \end{array} \right]$

(unterspezifiziert)

- **Verwendung in Syntaxanalyse**

→ Verwendung in **PSG-Regeln** zusammen mit **Constraintregeln** zum **Ausdruck von Abhängigkeiten** zwischen durch unterspezifizierte Merkmalsstrukturen repräsentierten **Kategorien**

→ nominales Agreement: **Beschränkung** der durch die PSG-Regel repräsentierten **Kombination** von Determinativ und Nomen **auf Übereinstimmung im AGR-Merkmal**:

$$\left[ \begin{array}{cc} \text{CAT} & NP \end{array} \right] \rightarrow \left[ \begin{array}{cc} \text{CAT} & DET \\ \text{AGR} & \boxed{1} \end{array} \right] \left[ \begin{array}{cc} \text{CAT} & N \\ \text{AGR} & \boxed{1} \end{array} \right]$$

- **Unifikation**

→ **Constraintregel**: entspricht Anweisung auf Durchführung von **Unifikation** zur **Feststellung der Vereinbarkeit**

→ nominales Agreement: Feststellung der Vereinbarkeit dieser **AGR-Teil-Merkmalstrukturen**:

$$\begin{bmatrix} \text{NUM} & \text{SG} \\ \text{GEN} & \text{MASK} \\ \text{CASE} & \text{NOM} \end{bmatrix} \sqcup \begin{bmatrix} \text{NUM} & \text{SG} \\ \text{GEN} & \text{MASK} \end{bmatrix} = \begin{bmatrix} \text{NUM} & \text{SG} \\ \text{GEN} & \text{MASK} \\ \text{CASE} & \text{NOM} \end{bmatrix}$$

# Sitzung 7: Unifikationsgrammatiken

- **Unifikationsbasierte Erweiterungen von CFGs**
  - Modellierung von **Agreement-, Rektions- und Subkategorisierungs-Constraints**
  - Modellierung von wortstellungsbezogenen Abhängigkeiten wie **Subjekt-Verb-Inversion** und *long distance dependencies*

- **Subkategorisierung**

→ Differenzierung der Klasse der Verben **nach Anzahl und Art ihrer Argumente** (z. B. auch nach abhängigen Sätzen)

→ **Subkategorisierungsprinzip**: Verb kann nur in **Umgebung** auftreten, die **seinem Subkategorisierungsrahmen entspricht**

→ mit **kontextsensitiven Regeln** oder als **Merkmalsconstraint** modellierbar

# Sitzung 8: Komplexe Satzkonstruktionen und Wortstellung

- **Wortstellung**
  - **strukturelle** Kodierung syntaktischer Funktion
  - **Positionierung** syntaktischer Einheiten
- **Wortstellungssyntax des Deutschen**
  - Verbstellungstypen: **V1, V2, VE**
  - **Verbstellungs-Split** kodiert Satzfunktion:

- V2 (Verbzweitstellung): **Aussagesatz**
- V1 (Verberststellung): **Aufforderungs-/Wunsch-/Fragesatz**
- VE (Verbendstellung): **Nebensatz**



- **Stellungsfeldermodell**

- **Lineares Modell** der Wortstellung des Deutschen, Analyse der Stellungsmöglichkeiten der Satzglieder
- **Einteilung in Felder**, ausgehend vom flektiertem Verbalkomplex als **Satzklammer**
- **diskontinuierliche Verbalphrase** kennzeichnend für Neu-hochdeutsch
- bei Verbzweitstellung kann **ein** beliebiges Satzglied ins Vorfeld gestellt werden (**Topikalisierung** bzw. **Fokussierung**)
- **Topik-Es** als Platzhalter wenn Vorfeld-Position unbesetzt
- **Wortstellungsregeln** der Anordnung von Satzgliedern im Mittelfeld, insbesondere '**Thema-vor Rhema**' (pragmatische Wortstellung)

- **Komplexe Satzkonstruktionen**

- **Einfache Sätze als Konstituenten von komplexen Sätzen**
- **Koordination = gleichrangige Verbindung**: Sätze bilden als **Ko-Konstituenten** einen komplexen Satz
- **Subordination = Einbettung** eines Satzes als **Satzglied des übergeordneten Satzes** (Matrixsatz)
- in **Dependenzanalyse**: **Verb des eingebetteten Satzes ist Dependent** von Verb des übergeordneten Satzes
- in **Konstituentenanalyse**: je nach Typ **andere Position im Syntaxbaum**: z.B. Objektsatz als Subkonstituente von VP
- **rekursive Einbettung**

- **Typen von eingebetteten Sätzen**
  - **Komplementsatz:** Subjekt- und Objektsatz
  - **Adverbialsatz**
  - **Attributsatz:** Relativsatz, adnominaler Substantivsatz
  - **Prädikativsatz**
- **Infinite Satzkonstruktionen**
  - können wie finite Sätze **als Satzglied auftreten**
  - **nicht-flektiert**, kein Subjekt
  - **Kontrolle durch Subjekt oder Objekt des Matrixsatzes**

- **Verbale Konstruktionen des Deutschen**
  - **Hilfs-und Modalverben (Auxiliare)** bilden mit **infinitiver Verbform** einen **Verbalkomplex**
  - **Auxiliar** als **linker Teil der Satzklammer**
  - **Satzklammer**: Aufteilung Satz in Felder → Vorfeld, Mittelfeld, Nachfeld
  - **Kopula** als **prädikatives Hilfsverb**, das mit einem Nomen, Adjektiv oder Satz eine **Eigenschaft** über das Subjekt oder Objekt prädiziert

# Sitzung 9: Parsing-Algorithmen

- 2 Klassen von **Parsing-Algorithmen: top-down / bottom-up**
  - top-down: **PREDICT + SCAN** (*Regelanwendung + Abgleich*)
    - probiert jede anwendbare Ersetzungsregel aus
    - im Problemfall: ***Backtracking*** *notwendig*
  - bottom-up: **SHIFT + REDUCE** (*Einlesen + Regelnw. rückwärts*)
    - verschiebt Token auf **Stapel** u. führt sie auf Regeln zurück

- **Vergleich** top-down vs. bottom-up:
  - Start der Analyse:
    - **Startsymbol** vs. **1. Wort der Eingabe**
  - Schwäche:
    - **strukturelle** vs. **lexikalische** Ambiguität
  - im Extremfall für beide **exponentielle Laufzeit**

- **Earley Parser: Top-Down-Parsing mit Extras**
  - 3 Operationen: **PREDICTION + SCANNING + COMPLETION**
    - *Voraussage*: wenn . vor Nichtterminal
    - *Überprüfung*: wenn . vor Terminal
    - *Vervollständigung*: wenn . letzte Position
  - **Zwischenergebnisse** werden in Datenstruktur (**Chart**) gespeichert (**Dynamische Programmierung**)
    - auch für ambige Grammatiken **maximal polynomielle Laufzeit**
  - **erweiterbar zu merkmalsbasiertem Parsing**
    - aber: Unifikation ist sehr **rechenaufwändig**

- **Statistisches Parsing:**
  - **nicht alle möglichen Ableitungen** werden ausprobiert, die **wahrscheinlichste** soll bestimmt werden
- per Hand geparste Sätze dienen als **Trainingsdaten**
- Eingabe wird in Merkmale umgewandelt (***Feature Extraction***)
- **Merkmalsvektoren** werden durch **gelernte Gewichte** auf eine **Wahrscheinlichkeitsverteilung** abgebildet
- die **Likelihood** der Trainingsdaten soll **maximiert** werden



# Sitzung 10: Unifikation

- **Subsumption:**
  - für Typen definiert durch die  $\sqsubseteq$ -Relation
  - bei Merkmalstrukturen muss es **alle Knoten der "allgemeineren" Merkmalstruktur auch in der spezifischeren geben (+ compatible Typen)**

- **Unifikation:**

- sowohl für Typen als auch Merkmalstrukturen **kleinste obere Schranke in der Subsumptionsbeziehung**

- für **Merkmale** zweischrittig:

- 1. **Identifikation äquivalenter Knoten**

- 2. **Unifikation ihrer Typen**

- **Bedingungen:**

- Pfade sind **Ketten von Merkmalen**
- Beschreibungen legen die **Menge von Merkmalstrukturen**, die sie erfüllen, **eindeutig fest**
- **Beschreibungen** werden im NLTK durch ihren **allgemeinsten Erfüller** ausgedrückt

# Sitzung 11: Statistische Syntaxmodelle

# 11.1 Probabilistische kontextfreie Grammatiken

- **Statistische Erweiterungen von CFGs**
  - mit **Abdeckung (*coverage*)** steigt **Anzahl an Ableitungen**
  - **statistische Modelle zur Disambiguierung**
  - **PCFG** (Probabilistische Kontextfreie Grammatik):  
**Gewichtung der CFG-Regeln mit Wahrscheinlichkeiten**
  - **Ranking** der Ableitungen nach ihrer **Wahrscheinlichkeit**

- **Eigenschaften von PCFGs**

- **Wahrscheinlichkeiten der Regeln zur Expansion von einem Nonterminal addieren sich zu 1**
- **Annahme Unabhängigkeit der Regel-Auswahl**
- **Wahrscheinlichkeit einer Ableitung: Multiplikation der Wahrscheinlichkeiten der in der Ableitung verwendeten Regeln**
- **Wahrscheinlichkeit einer Satzes: Summe der Wahrscheinlichkeiten seiner Ableitungen**

- **Abschätzung der Regelwahrscheinlichkeiten aus Trainingsdaten**
  - *supervised*: aus syntaktisch annotiertem Korpus (Treebank) über **relative Häufigkeiten der Expansionen eines Nonterminals (Maximum-Likelihood-Estimation)**
  - *unsupervised*, ohne Treebank: **Abschätzung durch wiederholtes Parsen eines Korpus und Anwendung von Expectation-Maximation-Algorithmus zur iterativen Verbesserung des statistischen Modells (Inside-Outside-Algorithmus)**

- **Probabilistisches Parsing**

→ **Suche der wahrscheinlichste Ableitung ( $T$ ) eines Satzes**

**( $S$ ):**  $\arg \max P(T|S)$

→ PCFG-Version des **Viterbi-Algorithmus** zum **effizienten Finden der wahrscheinlichsten Ableitung** mit dynamischer Programmierung



## 11.2 Statistische Dependenzmodelle

- **Statistische Dependenzmodelle**

- **Induktion von dependenzbasierten Syntaxmodellen** aus Dependency-Treebanks

- **Dependency-Treebanks = relationsannotiertes Korpus**

- Dependenzbäume können aus Konstituentenbäumen abgeleitet werden über **Kopfannotations- und Labeling-Regeln**

- entsprechend können auch Dependency-Treebanks (als Sammlungen von Dependenzbäumen) aus CFG-Treebanks wie der Penn-Treebank gewonnen werden

- **Übergangsbasiertes und Graph-basiertes** Dependenz-Parsing

- **Übergangsbasiertes Dependenz-Parsing**
  - **Stack-basierter Shift-Reduce-Parser**
  - **Auswahl des Übergangs** von einem Zustand (*Konfiguration von Stack, Buffer und erkannten Relationen*) zum nächsten **über Klassifikator**
  - **Klassifikator**: bildet Konfigurationen auf Übergänge ab
  - **trainiert anhand von Dependency-Treebank**
  - **Merkmale**: POS, Lemma, Token von obersten Elemente auf Stack, Buffer und den **Relationen** zwischen diesen Elementen

# Sitzung 12: Datengestützte Syntaxmodelle

## 12.1 Induzierte PCFG-Modelle

- **Induktion von PCFG-Grammatiken**
  - Modell trainieren anhand von **Treebank-Daten** (*supervised*)
  - **Extraktion von Regeln und Berechnung von Regelwahrscheinlichkeiten**
  - Aufbau von **empirischem Modell**
  - Form der induzierten Grammatik **abhängig vom Annotationschema der Treebank** (viele Regeln bei flachen Bäumen)

- **Normalisierung von CFGs**
  - **Chomsky Normalform**: u. a. zur **Reduktion der Regelmengen** von induzierten PCFGs
  - *Parent Annotation*: u. a. für *history-based* PCFGs
- **Evaluation von PCFGs**
  - **Übereinstimmung von Konstituenten** (PARSEVAL)
  - korrekte Konstituente: **gleiche Kategorie, gleiche Spanne**
  - **Recall, Precision, cross-brackets**

- **Lexikalisierte PCFGs**

- statistische **Modellierung lexikalischer Abhängigkeiten** wie PP-Attachment oder Subkategorisierung
- **Rücknahme** von PCFG-Annahme der **Unabhängigkeit einer Expansion von lexikalischer Information**
- **Annotation** syntaktischer Kategorie mit **lexikalischem Kopf**
- lexikalisiertes **Grammatikmodell** wird **sehr groß** (Regelvervielfachung)
- *sparse data* Problem mit ungesehenen Köpfen: **großes Trainingskorpus** und **Smoothing** notwendig

- **history-based PCFGs**

- statistische **Modellierung von Abhängigkeiten bzgl. des strukturellen Kontexts**

- **Rücknahme** von PCFG-Annahme der **Unabhängigkeit der Regelauswahl**

- **Annotation** syntaktischer Kategorie mit **Kategorie des Mutterknotens** (*parent annotation*)

- **Beispiel:** Subjekt-NP ( **$NP^S$** ) erweitert häufiger zu Pronomen als Objekt-NP ( **$NP^V$** )

## 12.2 Parsing mit neuronalen Netzen

- **Feed-Forward-Netzwerke (FFNs):**
  - FFNs sind eine Folge von linearen Abbildungen und nicht-linearen Aktivierungsfunktionen
  - Nichtlineare Transformationen (untere Schichten) machen die Daten zugänglich für einen linearen Klassifizierer (oberste Schicht)
  - Features für den linearen Klassifizierer werden von den unteren Schichten gelernt (kein Feature Engineering nötig)



- nichtlineare Abbildungen ermöglichen das Erlernen von nicht-linearen Zusammenhängen in den Daten
- Softmax-Regression ist typisches Modell für einen neuronalen Klassifizierer
- **Nachteile:**
  - benötigen tendenziell sehr große Datenmengen
  - haben viele Hyperparameter (schwierig zu optimieren)

- **Word Embeddings:**

- Alternative Wortrepräsentation zu One-Hot-Vektoren mit weniger Dimensionen
- Ähnlichkeit zwischen Wörtern wird berücksichtigt
- Kookkurrenz in unannotierten Texten ist Basis der meisten Embeddingmodelle (Distributional Hypothesis)

# Sitzung 13: Partielles Parsing

## 13.1 Partielles Parsing

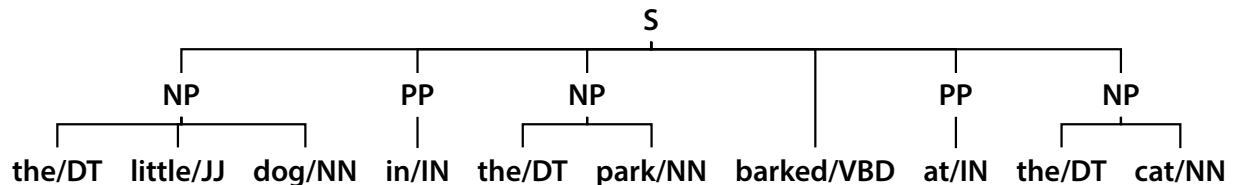
- **Partielles Parsing = Chunking**

→ Anwendungen wie Informationsextraktion oder *information retrieval* benötigen **keine syntaktischen Vollanalyse**

→ **unvollständige Analyse**: Finden nur der wichtigsten **Konstituenten** im Satz, primär **NP-, VP- und PP-Chunks**

→ **flache, nicht-hierarchische Analyse**: keine Verschachtelung

→ **Chunk = kleinere Einheit als vollständige Phrase**



- **Chunking mit regulärer Grammatik**
  - Beschreibung von **Muster von POS-Folgen** mit **regulären Ausdrücken**
  - **Chunking-, Chinking- und Split-Regeln**
- **kaskadierende Chunker**
  - **Loopen und Hintereinanderschalten von Chunk-Parsern**
  - **sukzessive Erzeugung hierarchisch aufgebauter Strukturen**

- **Lernbasiertes Chunking**

- **Klassifikation von Token-Sequenz** analog zu POS-Tagging ('*parsing as tagging*')

- **Lernen der Zuordnung von IOB-Tag zu Wort-POS-Tupel** aus **IOB-Chunk-getaggtem Korpus** (*supervised*)

- mögliche **Merkmale für *feature-extractor***:

- **POS-Tag und Wortform** des zu taggenden Tokens
    - **POS-Tag und Wortform der vorhergehenden und folgenden Tokens**
    - die bereits zugewiesenen **Chunk-Tags** der vorhergehenden Tokens

- **Evaluation von Chunkern**

- **Abgleich von Chunker-Output mit annotiertem Testkorpus**

- **Precision, Recall und F-score**

- **con112000-Korpus im NLTK als Chunk-getaggtes Korpus zum Testen und Trainieren**

## 13.1 Komplexität natürlicher Sprachen

- **Chomsky-Hierarchie:** Klassifizierung formaler Sprachen nach Stärke der **Regeleinschränkung** der sie erzeugenden Grammatik
- **kontextfreie Grammatik:** geeignet für Beschreibung der Phrasenstruktur natürlicher Sprache
- einige Syntaxformalismen sind **kontextsensitiv** (TAG, CCG) bzw. **rekursiv aufzählbar** (HPSG, LFG)



- **nicht-reguläre Konstruktionen in natürlicher Sprache: *center-embedding*-Rekursion:  $X \rightarrow \alpha X \beta$**
- auch **nicht-kontextfreie Konstruktionen: *cross-serial dependencies*** im Schweizerdeutschen
- solche **nicht-regulären Konstruktionen** sind aber **für die menschliche Sprachverarbeitung schwer zu verarbeiten** (aufgrund von *memory limitations*)
- Hinweise auf Berücksichtigung **statistischer Informationen** beim Parsing durch den Menschen: ***garden-path-Sätze***