Einführung in die Computerlinguistik

Morphologie III/ Syntax I

WS 2008/2009

Manfred Pinkal

Die NEA-DEA-Überführung

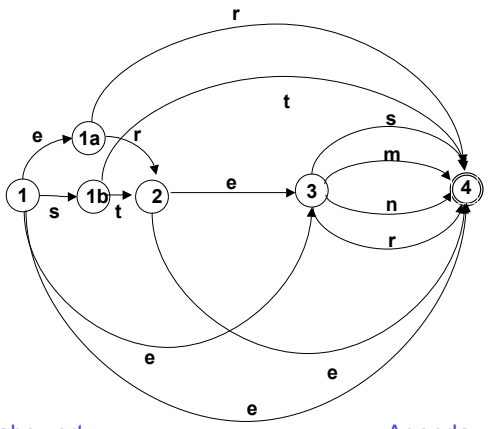
Der Algorithmus zur NEA-DEA-Überführung besteht aus drei Schritten:

1. Beseitigung von Mehrsymbol-Kanten
2. Beseitigung von ε-Kanten
3. Die „Potenz-Automaten"-Konstruktion

Schritt 3: Potenzautomaten-Konstruktion, Vorüberlegung

- Wir haben einen Algorithmus zur Pfadsuche am Beispiel des unbearbeiteten Adjektivendungs-Diagramms kennengelernt: „Tiefensuche mit Backtracking". Durch die Organisation der Agenda als Stapel/Stack („last in – first out") wird eine Alternative so weit wie möglich verfolgt; bei endgültigem Scheitern wird das System zurückgesetzt.
- Durch die Organisation der Agenda als Warteschlange (queue), bei der die Aufgaben in der Reihenfolge ihrer Generierung abgearbeitet werden („first in – first out"), erhalten wir Breitensuche. Die alternativen Pfade werden (quasi) parallel verfolgt.

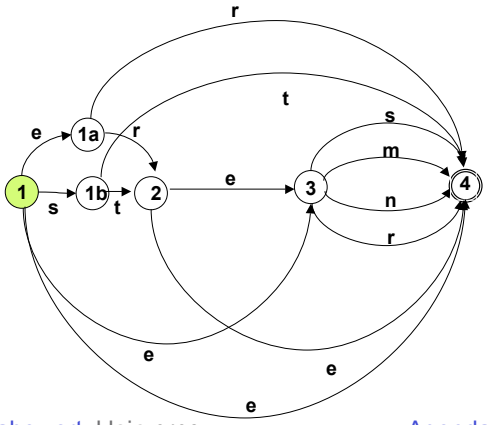
Pfadsuche als Breitensuche



Eingabewort:

Agenda: 1 -- klein eres

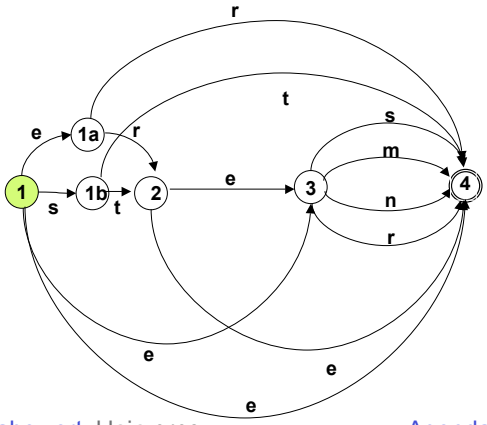
Pfadsuche als Breitensuche



Eingabewort: klein eres

Agenda: _____

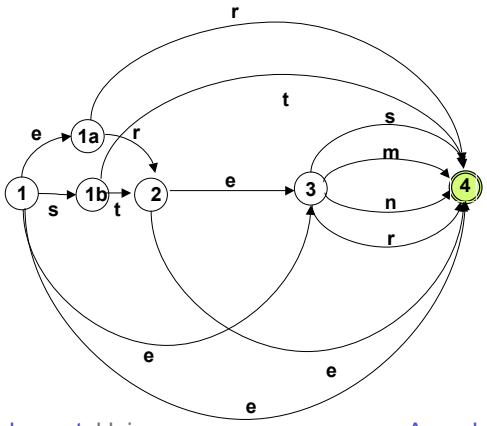
Pfadsuche als Breitensuche



Eingabewort: klein eres

Agenda: 1a -- klein eres
3 -- klein eres
4 -- klein eres

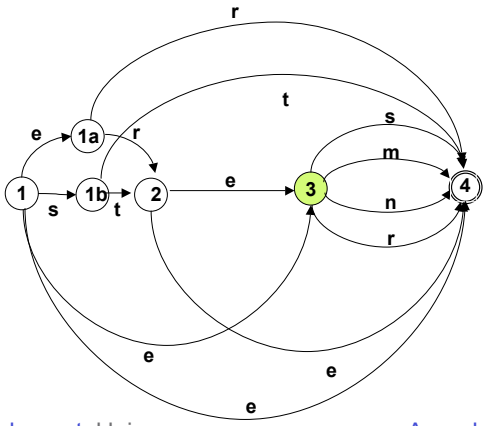
Pfadsuche als Breitensuche



Eingabewort: klein eres

Agenda: 1a -- klein eres
3 -- klein eres

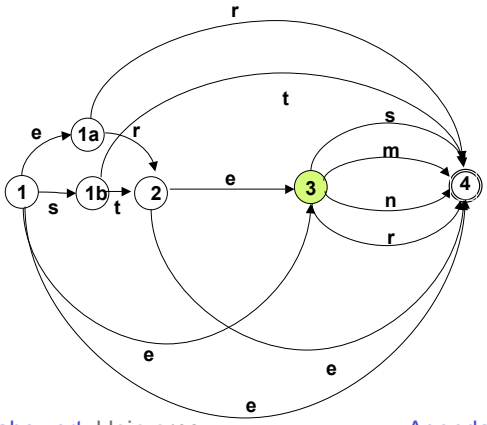
Pfadsuche als Breitensuche



Eingabewort: klein eres

Agenda: 1a -- klein eres

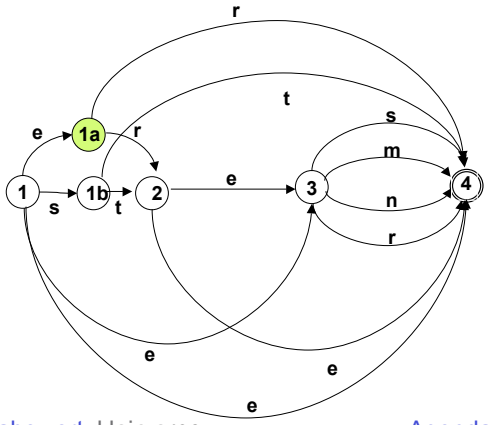
Pfadsuche als Breitensuche



Eingabewort: klein eres

Agenda: 4 -- klein eres
1a -- klein eres

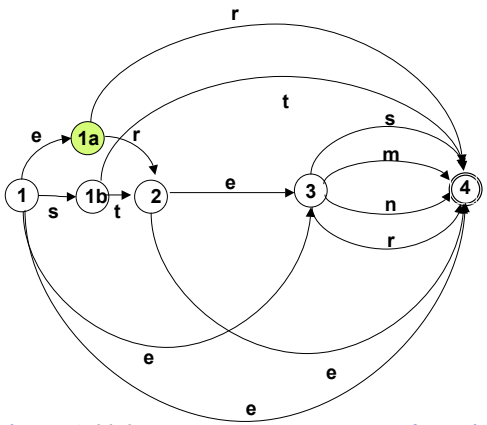
Pfadsuche als Breitensuche



Eingabewort: klein eres

Agenda: 4 -- klein eres

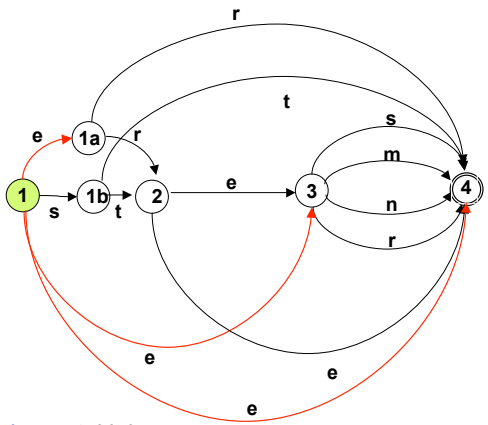
Pfadsuche als Breitensuche



Eingabewort: klein eres

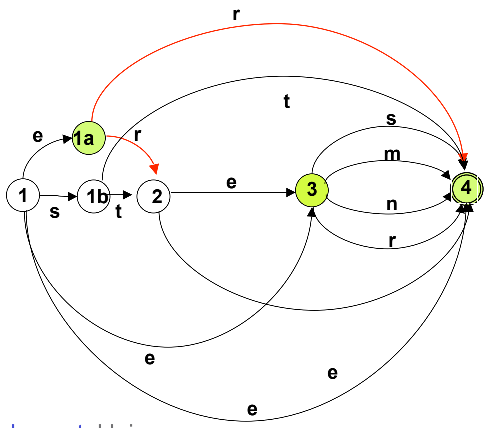
Agenda: 2 -- klein eres
4 -- klein eres
4 -- klein eres

Breitensuche „getaktet"



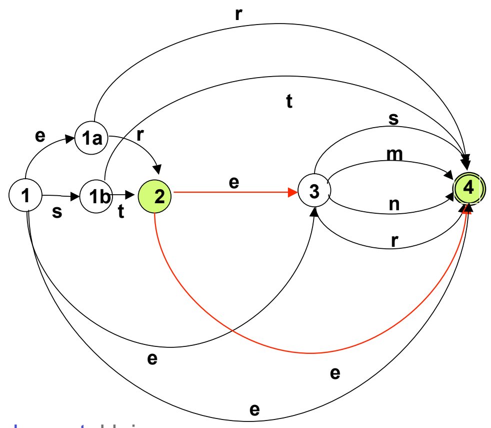
Eingabewort: klein eres

Breitensuche „getaktet“



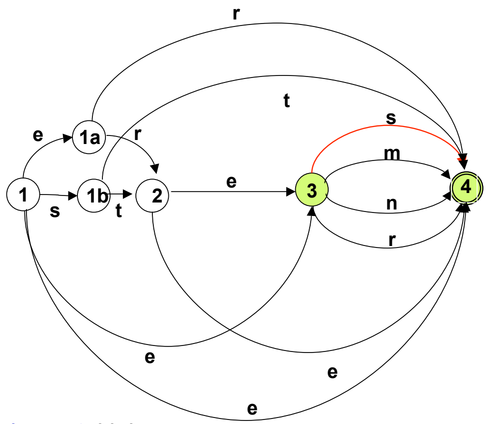
Eingabewort: klein eres

Breitensuche „getaktet“



Eingabewort: klein eres

Breitensuche „getaktet“



Eingabewort: klein eres

Schritt 3: Potenzautomaten-Konstruktion, Vorüberlegung [2]

- Wir können „getaktete“ Breitensuche in einem buchstabierenden NEA so beschreiben:
  - Wir ermitteln alle Zustände, die durch die Abarbeitung des ersten Eingabesymbols vom Startzustand aus erreicht werden können.
  - Wir ermitteln alle Zustände, die durch die Abarbeitung des zweiten Eingabesymbols von einem Zustand dieser Zustandsmenge erreicht werden können, usf.
  - Wenn die Zustandsmenge, die wir auf diese Weise nach Abarbeiten des kompletten Wortes w enthalten, einen Endzustand des NEA enthält, wird w akzeptiert.

Schritt 3: Potenzautomaten-Konstruktion, Vorüberlegung [3]

- Wir können diese "getaktete Suche" selbst mit einem endlichen Automaten beschreiben:
  - Zustände des neuen Automaten lassen sich als Mengen von Zuständen des NEA beschreiben. Am Beispiel: Nach Abarbeiten des ersten Symbols „e“ befindet er sich in dem Zustand, dass es die Zustandsmenge des NEA {1a, 2, 4} als mögliche aktuelle Zustände erkannt hat.
  - Wenn die Eingabekette abgearbeitet ist, und der Automat sich in einem Zustand befindet, der einen Endzustand des NEA enthält, ist die Eingabe akzeptiert.
  - Die „möglichen Zustände“ des NEA, die sich durch ein bestimmtes Eingabe-Symbol erreichen lassen, sind eindeutig definiert. Der neue Automat ist also ein DEA.

Schritt 3: Potenzautomaten-Konstruktion: Die Definition

Der Potenzautomat zum buchstabierenden NEA
$A = \langle K, \Sigma, \Delta, s, F\rangle$ ist der DEA A':

$A' = \langle K', \Sigma, \delta, s', F'\rangle$ mit:

- $K' = \wp(K)$ (die Potenzmenge der Zustandsmenge des NEA)
- $s' = \{s\}$
- $\delta(p', a) = \{q \mid$ es gibt $p \in p'$ und $\langle p,a,q\rangle \in D\}$ für jedes $p' \subseteq K$, $a \in S$
- $q' \in F'$ gdw. $q' \cap F \neq \emptyset$

Praktisches Vorgehen

Der Potenzautomat A' zu $A = \langle K, \Sigma, \Delta, s, F\rangle$ hat $2^{|\Delta|}$ Zustände. In der Regel sind viele dieser Zustände unerreichbar (vom Startzustand {s} aus) und deshalb funktionslos.

Praktisches Konstruktionsverfahren:

Beginne mit {s}, berechne die Übergangsfunktion für {s}, für alle direkt von s erreichbaren Zustände usw., bis keine neuen erreichbaren Zustände hinzukommen.

Beispiel: DEA für Adjektiv-Endungen

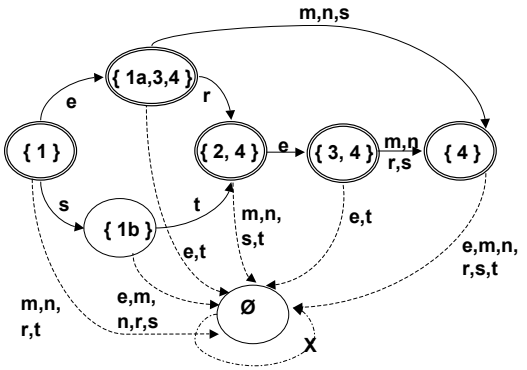
- Grundlage: der buchstabierende Automat
  $A = \langle\{1,1a,1b,2,3,4\}, \{e,m,n,r,s,t\}, \Delta, 1, \{1,4\}\rangle$,
  Δ wie im Diagramm Folie 42

- Potenzautomat ist $A' = \langle K', \Sigma, \delta, s', F'\rangle$
  mit $K' = \wp(K)$
  $s' = \{s\}$
  $F' = \{q' \in K' \mid 1 \in q' \text{ oder } 4 \in q'\}$
  δ s. Übergangstabelle nächste Folie

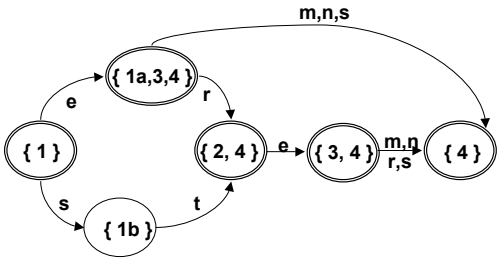
DEA für Adjektiv-Endungen, Übergangstabelle

δ:	e	m	n	r	s	t
{1}	{1a,3,4}	∅	∅	∅	{1b}	∅
{1a,3,4}	∅	{4}	{4}	{2,4}	{4}	∅
{1b}	∅	∅	∅	∅	∅	{2,4}
{2,4}	{3,4}	∅	∅	∅	∅	∅
{3,4}	∅	{4}	{4}	{4}	{4}	∅
{4}	∅	∅	∅	∅	∅	∅
∅	∅	∅	∅	∅	∅	∅

Das Diagramm



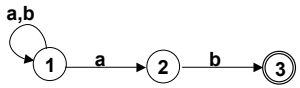
Das Diagramm, vereinfacht



Potenzautomatenkonstruktion, ein weiteres Beispiel

NEA A= < {1,2,3}, {a,b}, Δ,1, {3} >

Δ gegeben durch:



DEA

A‘ = < $\wp$ ({1,2,3}), {a,b}, δ, {1}, F‘>)

F´= {{3}, {1,3}, {2,3}, {1,2,3}}

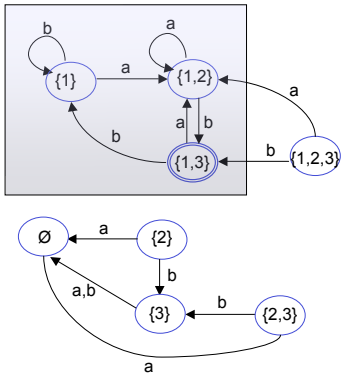
Potenzautomatenkonstruktion, Beispiel 2: Die Übergangstabelle

q	δ(q, a)	δ(q, b)
{1}	{1,2}	{1}
{2}	Ø	{3}
{3}	Ø	Ø
{1,2}	{1,2}	{1,3}
{1,3}	{1,2}	{1}
{2,3}	Ø	{3}
{1,2,3}	{1,2}	{1,3}
Ø	Ø	Ø

Potenzautomatenkonstruktion, Beispiel 2: Das Zustandsdiagramm

Nur ein Teil der Zustände ist vom Startzustand aus erreichbar.

Die übrigen Zustände sind funktionslos.



Potenzautomatenkonstruktion, Beispiel 2: Die Übergangstabelle

q	δ(q, a)	δ(q, b)
{1}	{1,2}	{1}
{2}	Ø	{3}
{3}	Ø	Ø
{1,2}	{1,2}	{1,3}
{1,3}	{1,2}	{1}
{2,3}	Ø	{3}
{1,2,3}	{1,2}	{1,3}
Ø	Ø	Ø

Morphologiesysteme

- Flexionsmorphologie: Lemmatisierung/Stemming
  - *veranstalt+et, Veranstaltung+en*
- Ableitungs-/Derivationsmorphologie
  - *Veranstalt+ung, un+glaubwürdig*
- Komposita-Zerlegung
  - *Fach+veranstaltung, glaub+würdig*

Morphologiesysteme

- Flexionsmorphologie: Lemmatisierung/Stemming
- Ableitungs-/Derivationsmorphologie
- Komposita-Zerlegung

Lemmatisierung/Stemming

- Rückführung flektierter Formen auf Stämme, erlaubt je nach morphologischer Struktur der Sprache Reduktion der Lexikongröße (Verhältnis Wortstämme/Lemmata : Wortformen) von 1:2 (Englisch), 1:5 (Deutsch), 1:200 und mehr (Türkisch, Finnisch)
- Ermittlung von grammatischen Merkmalen, die für die syntaktische Analyse nötig sind
- Methode: Endliche Automaten bzw. Transduktoren (s. Übungsaufgabe) + Flexionsklassen-Information im Lexikon

Wortbildungsmorphologie: Ableitung/Derivationsmorphologie

- Derivationsmorphologie
  - erschließt ein größeres Vokabular für die Rechtschreibprüfung (Vervielfältig+ung)
  - Liefert über das Suffix Wortart- und Flexionsklasseninformation als Grundlage für die syntaktische Verarbeitung (feminines, schwach flektiertes Substantiv)
  - Macht den Stamm für die Informationssuche sichtbar

- Derivationsmorphologie ist in verschiedener Hinsicht unsystematisch:
  - viele Ableitungspräfixe und -suffixe sind semiproduktiv:
  - viele Ableitungen sind semantisch "nicht transparent": Sie haben eine konventionelle, lexikalisierte Bedeutung, die mit der Bdeutung des Stammworts nicht in systematischer Beziehung steht.
  - Beispiele:
    - die *Lesung* bezeichnet den Akt des Vorlesens,
    - die *Singung* ist ist unmöglich
    - die *Vorlesung* gibt es, bezeichnet aber nicht den Akt des Vorlesens,
    - die *Schreibung* nicht den Akt des Schreibens

Morphologiesysteme

- Lemmatisierung/Stemming
- Ableitungs-/Derivationsmorphologie
- Komposita-Zerlegung

Kompositazerlegung

- Kompositazerlegung ist unerlässlich (mindestens im Deutschen), z.B. für
  - Rechtschreibprüfung
  - als Grundlagen für die syntaktische Verarbeitung
- Kompositazerlegung vervollständigt (ebenso wie Derivationsmorphologie) den Informationszugriff: Suche nach *Rentenversicherung* soll auch *Angestelltenrentenversicherung, Privatrentenversicherung etc.* finden.

Anforderungen an Morphologiesysteme

- Korrektheit
- Vollständigkeit / Abdeckung (engl. „coverage")
- Effizienz

Dies sind elementare Anforderungen an alle sprachverarbeitenden Systeme - im Prinzip an alle Computersysteme

Korrektheit

- Anforderungen:
  - Nur gültige Wortformen werden analysiert
  - Korrekte Zerlegung in Morpheme
  - Korrekte grammatische Merkmals-Information
- Flexionsmorphologie: Typischerweise unproblematisch, Korrekt, wenn die Flexionsklassen im Lexikon korrekt angegeben sind.
- Derivationsmorphologie:
  - Tendenziell Übergenerierung (Semiproduktivität)
  - Tendenziell semantisch irreführende Identifikation von Stämmen
- Kompositazerlegung:
  - Übergenerierung ist ein massives Problem
  - ... wenn sie nicht durch Zusatzmechanismen behoben wird (Blockierungslisten, statistische Gewichtung)

Übergenerierung: Beispiele aus der Praxis

- Ein klassiches Beispiel aus der maschinellen Übersetzung (Systran, um 1980)
  - Barbarei
  - -> nightclub nightclub egg
  - Bar|bar|ei
- Ein Beispiel aus der Rechtschreibkonversion (Corrigo, um 2000)
  - Hufeisenniere
  - -> Hufeisennniere
  - Huf|ei|senn|niere

Abdeckung

- Aktuelle Morphologiesysteme haben eine gute bis sehr gute Abdeckung (s. z.B. Word-Rechtschreibung)
- Zur Terminologie:
  - Falsche Positive sind unmögliche Wortformen, die fälschlich akzeptiert werden (Korrektheitsfehler)
  - Falsche Negative sind zulässige Wortformen, die nicht akzeptiert werden (Abdeckungsfehler)
- Abdeckung und Korrektheit allein sind für sich genommen keine guten Bewertungskriterien:
  - Man kann Korrektheit billig auf Kosten der Abdeckung erreichen und umgekehrt.
  - Ziel: Zuverlässigkeit bei gleichzeitig großer Abdeckung

Effizienz

- Grundsätzlich: Morphologische Analyse benötigt lineare Zeit in Abhängigkeit von der Länge der Eingabe.
- Gute Morphologiesysteme liegen im Bereich von ca. 1 ms pro Wortform (auf normalem PC)
- Durch Vorverarbeitung, Zwischenspeichern von Analysen, Indexierung etc. lässt sich für größere Dokumente die Zeit pro Textwort auf den unteren μs-Bereich drücken.
- Das ist
  - exzellent für Online- und Offline-Rechtschreibkorrektur
  - akzeptabel für begrenzte Datensammlungen (große Textkorpora, Firmen-Intranet etc.)
  - zu langsam fürs Web

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]

- Wie finden Sie stattdessen das angehängte Bild? Das ist ein Foto, das im Rahmen des TALK-Projektes entstanden ist, uns gehört, und von BMW schon freigegeben war. Außerdem vermittelt es besser den Bezug zur Forschung.

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 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.