
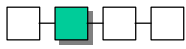
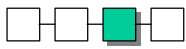



Ein hybrider Ansatz zur Analyse gesprochener Spontansprache

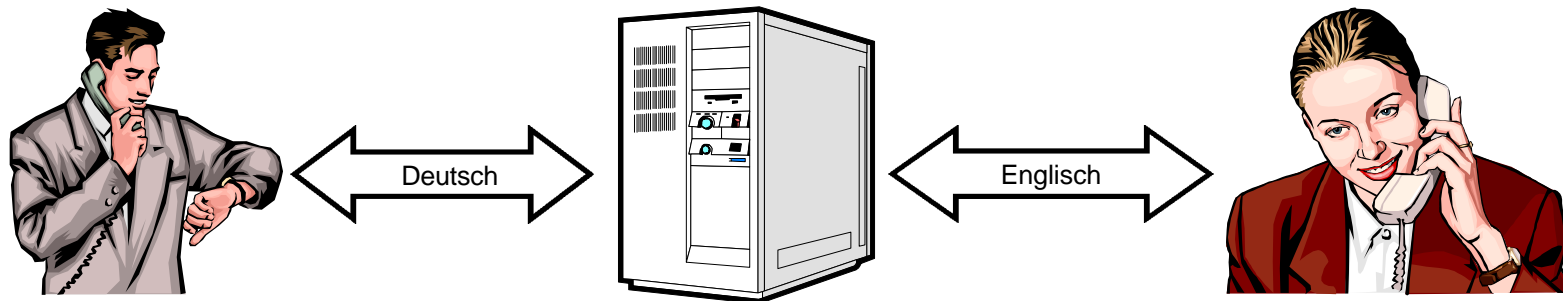
Tobias Ruland

4. April 2001

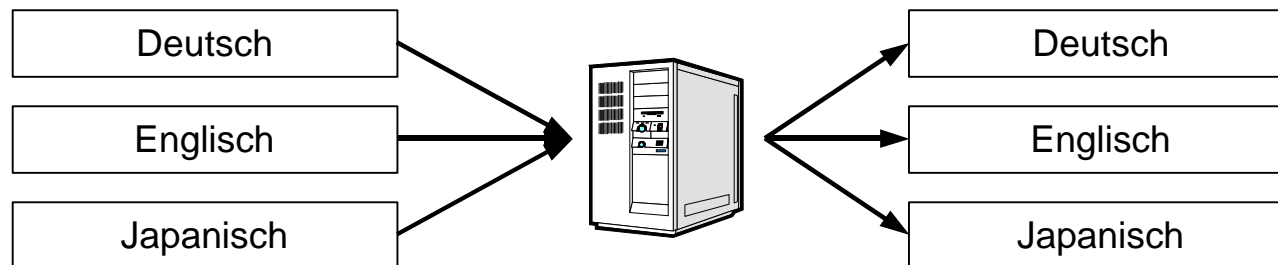
Gliederung des Vortrags

1. Einbettung der Arbeit und Problembeschreibung 
2. Frühere Ansätze zur Problemlösung 
3. Beschreibung der vorgestellten Arbeit 
4. Ergebnisse der Arbeit und Ausblick 

Die vorgestellte Arbeit entstand im Rahmen des BMBF-Verbundprojekts
Verbmobil

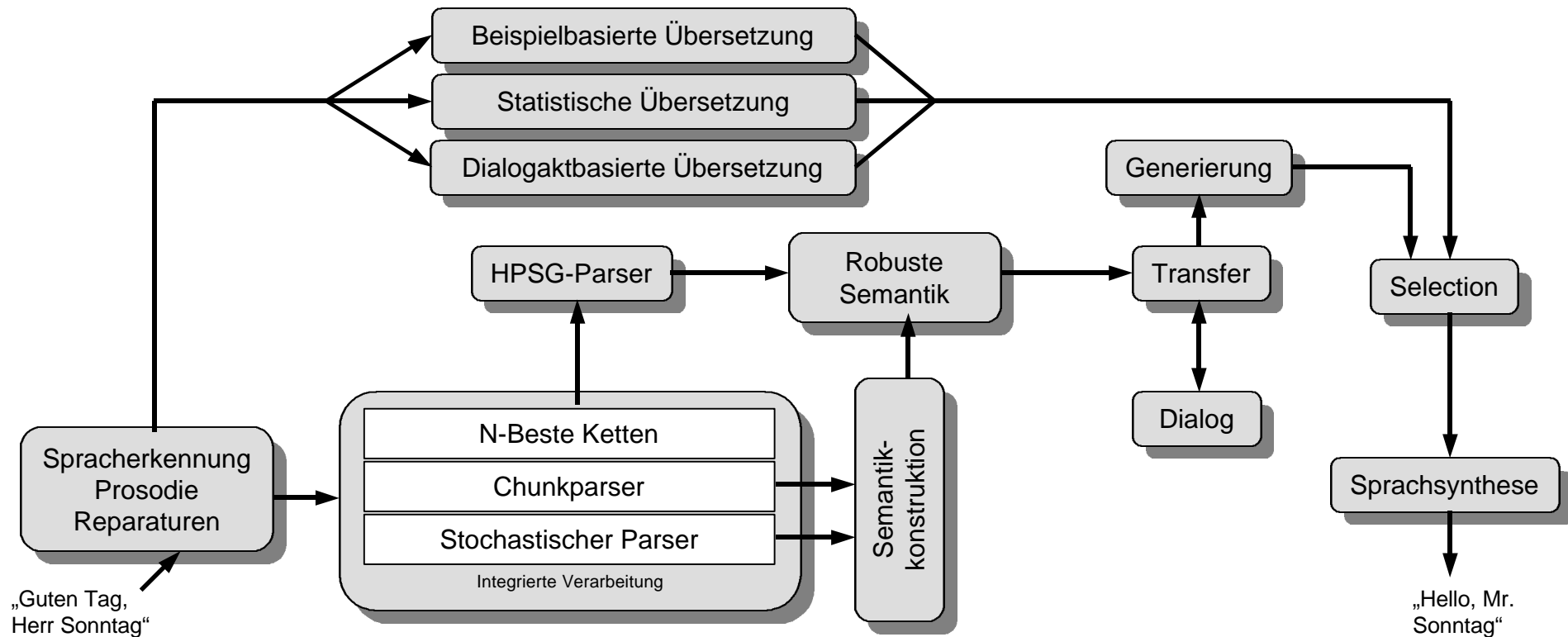


Verbmobil



Verbmobil übersetzt Spontansprache in gesprochene Sprache

In *Verbmobil* arbeiten mehrere Analysemodule parallel an der Lösung einer Aufgabe



Die „beste“ Analyse wird präferiert und deren Übersetzung verbalisiert.

Die spontansprachlichen Eingaben sind außerordentlich schwierig zu analysieren

Schön, passt mir auch.

Das wär' der der fünfte, sechste.

Ich würde fast vorschlagen, nachdem wir eigentlich Zeit haben, ist es gemütlicher, im Zug zu fahren

Ach so, naja, Ausstellungen, Sie wissen ja.

Das klingt und zumal ich glaube, das ist zentral.

Machen wir's dann?

Oder so oder ja, neun Uhr.

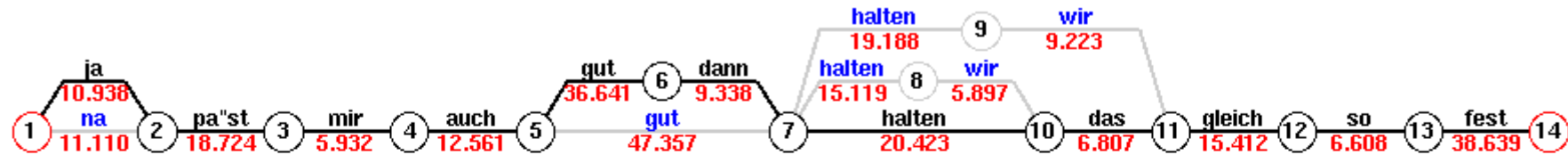
Bedank' mich, freue mich, auf Wiedersehen.

Genau, da buch' ich die Flüge und.

Das Hotel am neben dem Bahnhof, Sie erinnern, können Sie sich erinnern?

- Schwierigkeiten:
- Spontansprachliche Phänomene wie Reparaturen, Gedankensprünge etc.
 - „Korrektheit“ der Eingabe, z.B. Satz-/Wortabbrüche, zerstörte Satzstruktur
 - bestimmte grammatische Phänomene, z.B. Parenthesen
 - gesprochen vs. geschrieben, z.B. Satzzeichen, Groß-/Klein-Schreibung
 - etc. etc. etc.

Beispiel eines *Verbmobil*-Worthypothesengraphen



Im Allgemeinen sind Wortgraphen viel dichter und enthalten mehr Fehler.

Unterschiedliche Analysemodule erfüllen zentrale Forderungen an die Analyse von Spontansprache unterschiedlich gut

	HPSG-Parser	Chunkparser	Stoch. Parser	Dialogaktbas. Übersetzung	Stat. Übersetzung	Beispielbas. Übersetzung
Robustheit ggü. Erkennungsfehlern	-	○	○	+	+	+
Auffinden gramm. korrekter Äußerungen	+	-	○	-	-	-
Berücksichtigung prosodischer Information	○	○	○	○	○	○
Semantische Plausibilität der Analyse	+	-	○			
Verarbeitungsgeschwindigkeit	-	+	○	+	○	+
Abdeckung der Domäne	○	○	○	○	○	○
Abdeckung von OOD-Äußerungen	+	+	○	-	○	-
Abdeckung von OOV-Äußerungen	-	○	-	+	-	+

Die vorgestellte Arbeit steht in Beziehung zu einer Reihe anderer Arbeiten

1. Stochastisches LR Parsing natürlicher Sprache
2. Parsing gesprochener Sprache
3. Automatisches Lernen regelbasierter Parser

Einige wichtige Arbeiten werden im Folgenden kurz dargestellt.

Parsers induced from corpora are of interest both as simply exercises in machine learning and also because they are often the best parsers obtainable by any method.
[E. Charniak, 1997]

- *Wright & Wrigley, 1991*
 - PCFG-äquivalent (\Rightarrow kein Regelkontext)
 - Modell berechnet Übergangswahrscheinlichkeiten von einem „shiftbaren“ Stack zum nächsten
 - benutzt Strahlensuche zur Suchraumbeschränkung
- *Briscoe & Carrol, 1993*
 - probabilistisches Modell zur „Unterstützung“ einer Unifikationsgrammatik
 - $P(T) \approx \prod P(l_i, a_i, s_i | s_{i-1})$
 - manche Bewertungen gehen doppelt in die Berechnung ein
- *Inui et al., 1997*
 - $P(T) \approx \prod P(l_i, a_i, \sigma_i | \sigma_{i-1})$
 - bessere Ergebnisse als WW91 und BC93
 - immer noch minimaler LR-Parser-Kontext

In den 90er Jahren gab es eine größere Zahl von Arbeiten zum Parsing gesprochener Sprache

- *Schmid, 1994*
 - nicht-probabilistisches LR-Parsing von *Verbmobil*-Worthypothesengraphen
 - A*-Suche
 - Parser benutzt Unifikationsgrammatik (Siemens Trace & Unification Grammar)
- *Weber, 1994*
 - probabilistisches Chartparsing von *Verbmobil*-Worthypothesengraphen
 - Zusammenfassen mehrerer „Wissensquellen“ in einer Bewertung
 - LR-inkrementelles Verfahren mit Strahlensuche
- *Lavie, 1996*
 - probabilistisches LR-Parsing von N-Besten-Listen
 - Verwendung einer Unifikationsgrammatik
 - probabilistisches Modell sehr ähnlich Briscoe & Carrol, 1993

Zum automatischen Lernen regelbasierter Parser gibt es in den letzten Jahren eine Reihe interessanter Arbeiten

- *Brill, 1993*
 - interessanter TBL-Ansatz für POS-Tagging
 - einfacher Zweistufen-Parser für WSJ-Sätze (1000 Trainingsäußerungen)
- *Zelle, 1995*
 - lernt eine prozedurale Grammatik (DCG) mittels ILP
 - getestet auf 525 ausgewählten ATIS-Äußerungen
- *Hermjakob, 1997*
 - lernt einen deterministischen Shift-Reduce-Parser
 - Entscheidungsbaumverfahren
 - sehr aufwendiges Verfahren, großer Zustandsraum

Das verwendete Modell baut auf bewährten Vorgehensweisen auf

1. Gesucht: $P(T|W) = \alpha \cdot P(T) \cdot P(W|T)$ [IST97]

2. $P(W|T)$ lässt sich auf viele Arten abschätzen, häufig

$$P(W|T) \approx \prod P(w_i|l_i)$$

3. $P(T) = ???$

- [WW91]: PCFG-Bewertung von T

- [BC93]: $P(T) \approx \prod P(l_i, a_i, s_i | s_{i-1})$

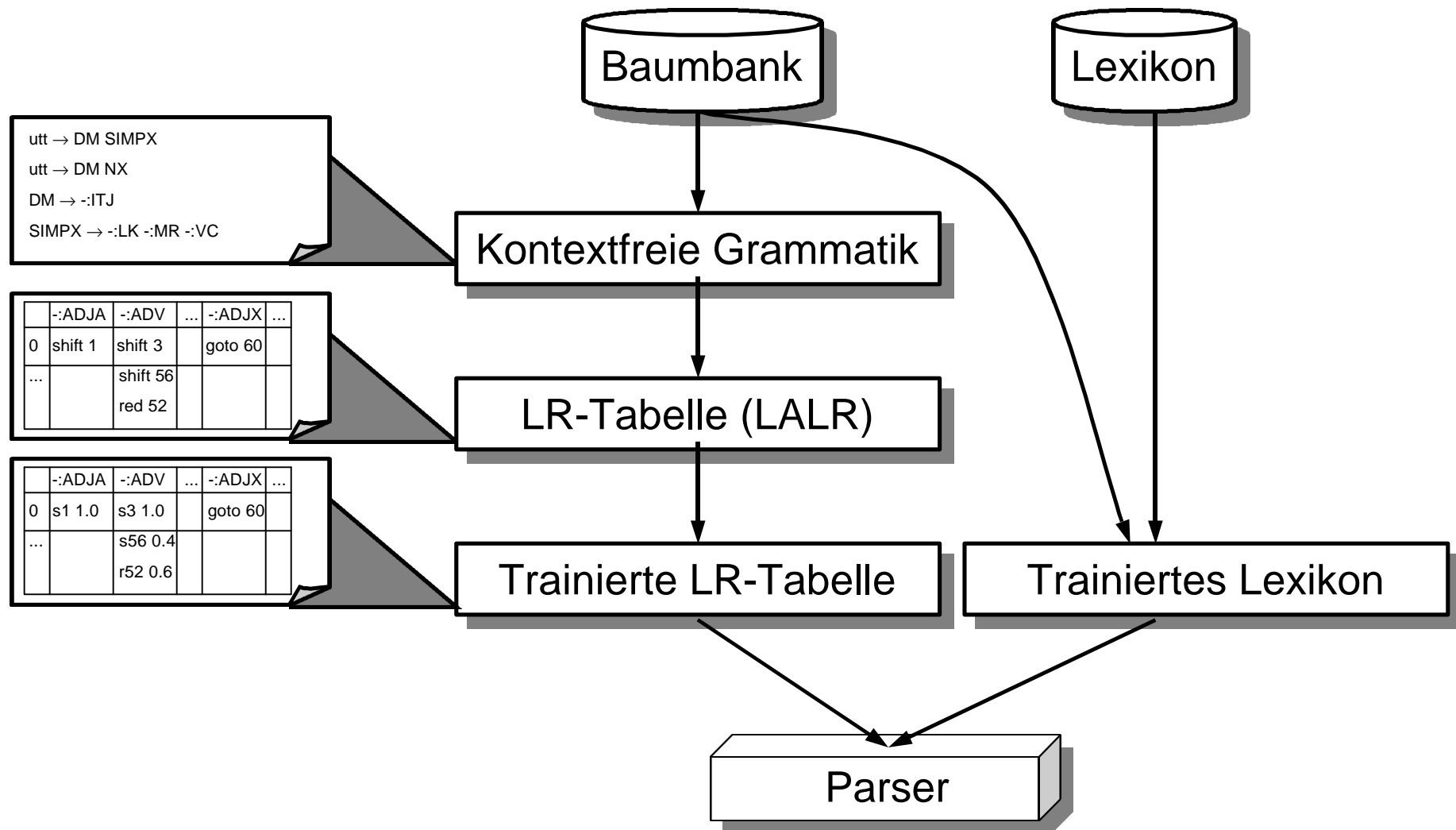
- [IST97]: $P(T) \approx \prod P(l_i, a_i, \sigma_i | \sigma_{i-1})$

$$P(T) \approx P(T|L) = \prod P(l_j, a_j, s_{d,j} | s_{d,j-1}, l_{d,j-1}, \dots)$$

d.h. Modell ähnlich BC93, jedoch mit größerem Kontext. Letzlich müssen wir $P(a_{d,j} | k_{d,j})$ abschätzen.

l: Look-Aheads s: LR-Zustände
a: Parseraktion σ : LR-Stacks
k: Parsingkontext T: Parsebaum
W: Eingabe/Wortfolge

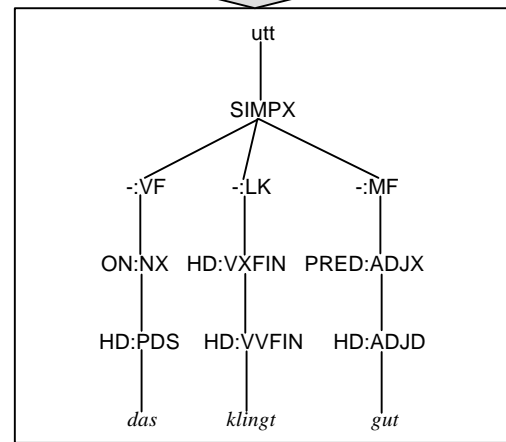
Der probabilistische LR-Parser bezieht sein Wissen aus mehreren Quellen, welche stufenweise erzeugt werden



Eine Analyse besteht aus einer Reihe von Parseroperationen

das klingt gut

shift, reduce 1838, reduce 1232, shift, reduce 1478, reduce 206, shift, reduce 1934, reduce 976, reduce 1198, reduce 122



$P(\text{shift}) \approx P(\text{shift}|K) \cdot P(\text{Tag|Wort})$

$P(\text{reduce}_N) \approx P(\text{reduce}_N|K)$

$$P(\text{Baum}|\text{Satz}) \approx P(O_1) \cdot P(O_2) \cdot \dots \cdot P(O_n)$$

O_i: i-te Parseroperation

K: Kontext

Die Abschätzung der Operationswahrscheinlichkeiten ist nicht trivial

Aufgabe: schätze $P(a|k)$

Problem: für jedes Feld der LR-Tabelle soll gelten: $\sum_i P(a_i|k) = 1$

Haken: es gibt unendlich viele verschiedene k 's, aber nur begrenzt viel Trainingsmaterial, begrenzt viel Speicherplatz und begrenzt viel Zeit

a: Parseraktion
k: Parsingkontext

Durch Smoothing lassen sich die $P(a|k)$ pragmatisch abschätzen

Ein Kontext K besteht aus einer Menge elementarer Subkontexte $\{k_1, k_2, \dots, k_n\}$.

Der Entwickler wählt daraus Teilmengen K_1, K_2, \dots, K_m aus.

Approximiere (durch Interpolation)

$$P(a|K) \approx \sum \alpha_i P(a|K_i)$$

Auf *Verbmobil*-Daten verwendeten wir u.a.

- | | |
|-------|--|
| K_1 | Aktionen-Unigramm |
| K_2 | LR-Zustand |
| K_3 | K_2 & Look-Ahead |
| K_4 | K_3 & Phrasenkopf d. obersten LR-Stackelements |
| K_5 | LR-Zustand & folgendes Wort (Look-Ahead-Wort) |

Das Durchsuchen des Wortgraphen bewerkstelligt eine abgewandelte A*-Suche

1. $H = \text{berechne_A_Restkosten}(W)$
2. Initialisiere die leere sortierte Agenda A_0
3. $A_1 = P.\text{start}(A_0, W, H, n_0)$
 $i = 1$
4. Solange $\text{out}(\text{last_node}(\text{top}(A_i))) \neq \{\}$ und $\neg P.\text{accept}(\text{top}(A_i))$ und $\neg \text{empty}(A_i)$:
$$A_{i+1} = P.\text{expand}(A_i)$$
5. Falls $\neg \text{empty}(A_i)$ gib als Ergebnis $\text{stack}(\text{top}(A_i))$ aus, sonst Fehler

A_i : Agenda

H : mit Restkosten annotierter Wortgraph

W : Wortgraph

n_0 : Startknoten

P : Parser

Die Metrik kombiniert drei Einzelbewertungen

1. Akustik, d.h. Bewertung der verarbeiteten Worthypothesen durch den Spracherkenner
2. Sprachmodell, d.h. Bewertung durch *Verbmobil*-Trigramm
3. Parsing, d.h. Bewertung einer Folge von Parseroperationen

d.h.

$$\text{Score} = \alpha_0 AS + \alpha_1 LS + \alpha_2 \frac{PS}{PO} + RK$$

In *Verbmobil*:

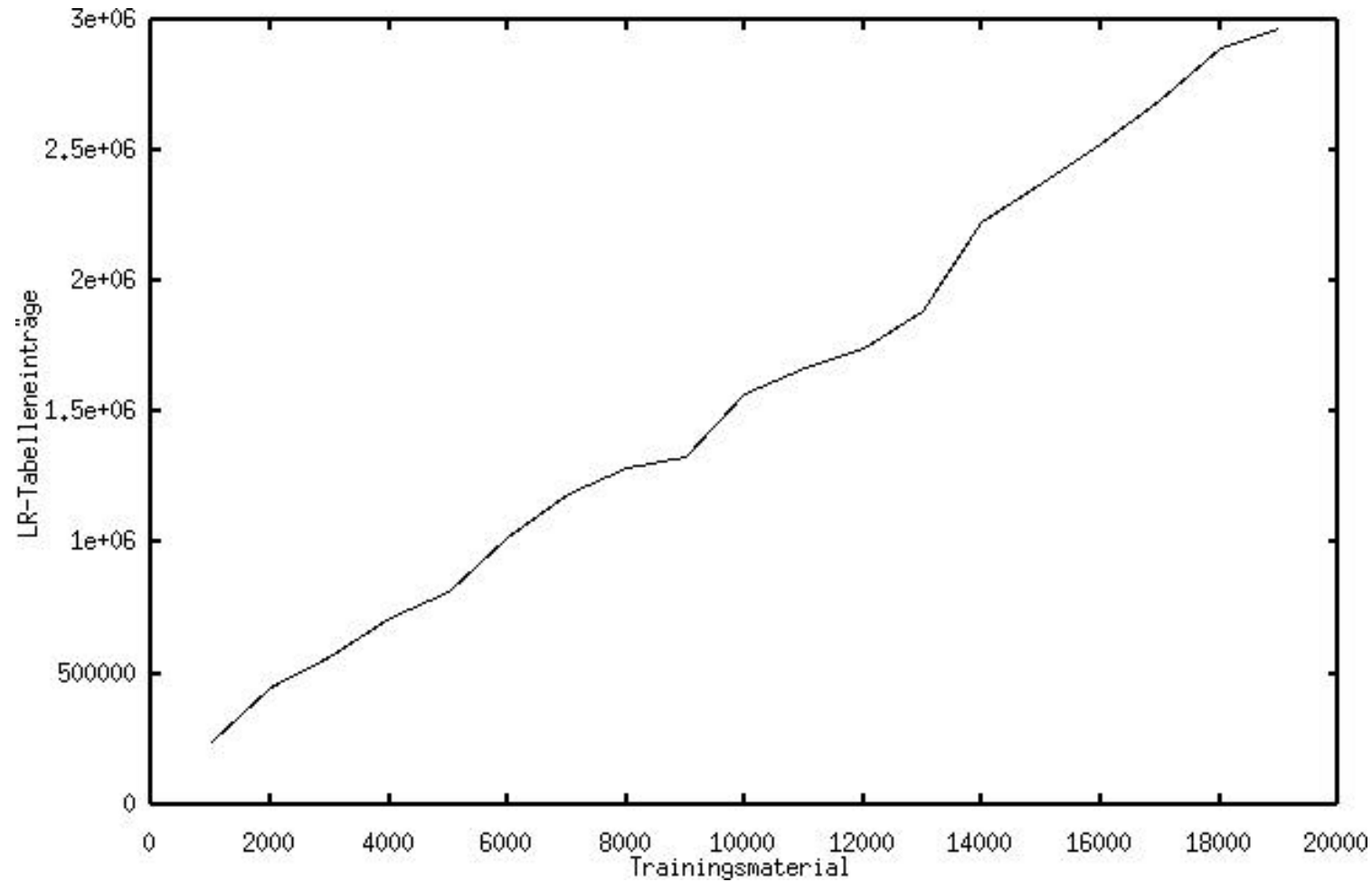
$$\alpha_0 = \alpha_1 = 1 \quad \alpha_2 \approx 0,1$$

Die Suche findet also beste grammatisch korrekte Kette und die dritte Komponente selektiert darauf den bestbewerteten Baum

Die Suche des Parsers wird durch mehrere Faktoren beschränkt

1. Die Größe der LR-Tabelle wird beim Training eingeschränkt
2. Agendaeinträge werden nur dann neu in die Agenda eingetragen, wenn ihre Bewertung einen Maximalabstand vom bestbewerteten Eintrag nicht übersteigt (Strahlensuche)
3. Parser „darf“ nur eine festgelegt Zahl von Operationen ausführen.

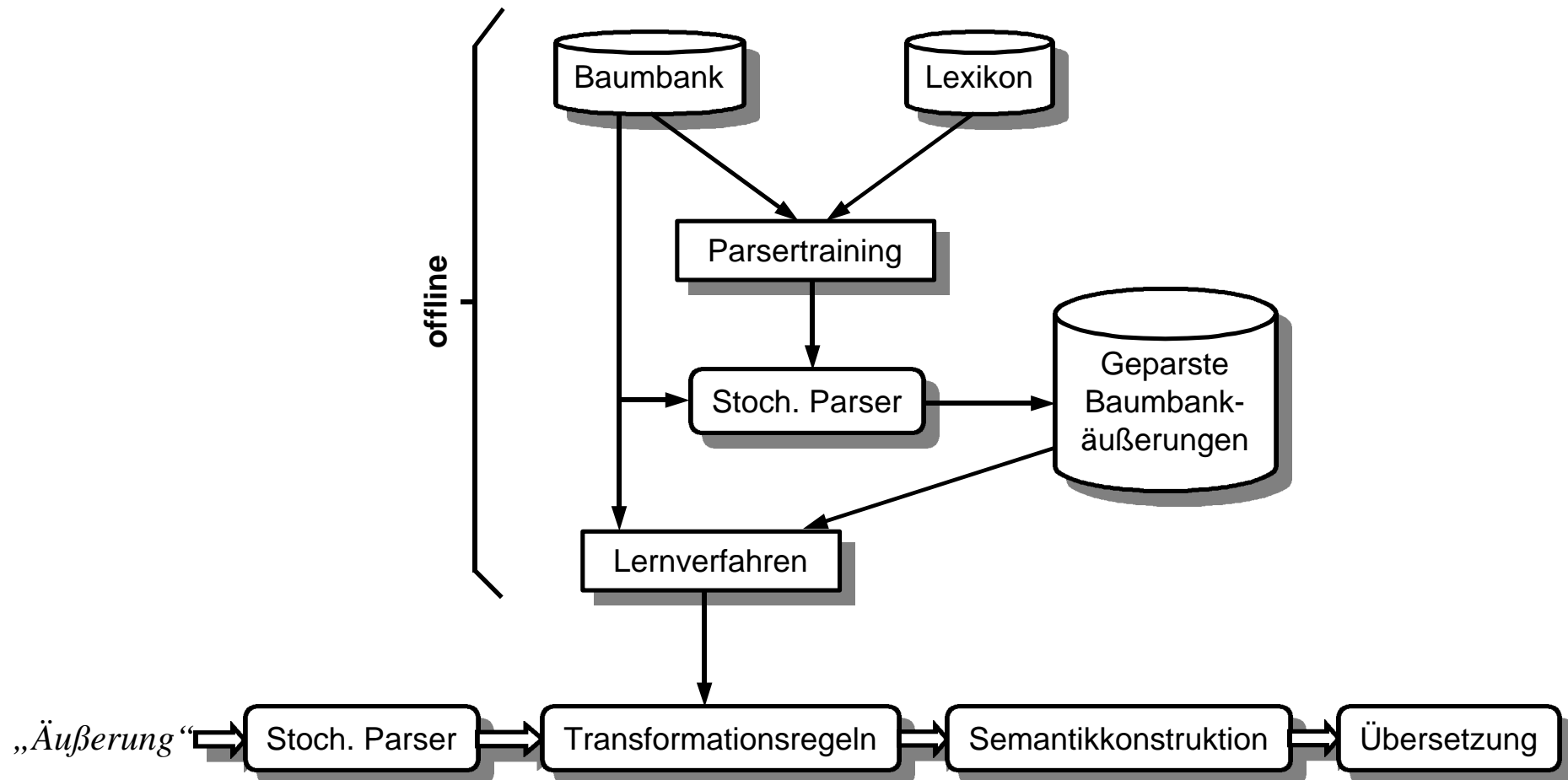
Die Größe der LR-Tabelle scheint linear mit der Zahl der Trainingsäußerungen zu wachsen



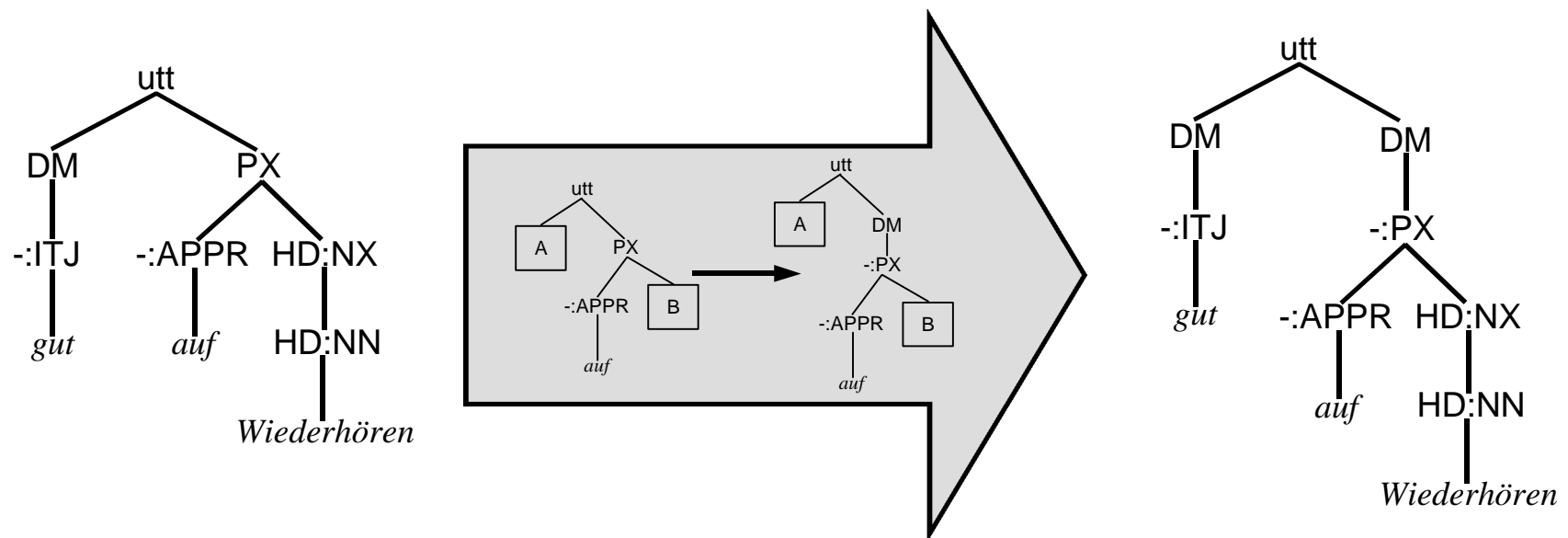
Dies macht Techniken zur Begrenzung der Tabellengröße notwendig!

Die Anwendung der Transformationsregeln benötigt im Laufzeitsystem nur minimale Ressourcen

Einbettung des Lernverfahrens und der gelernten Transformationsregeln



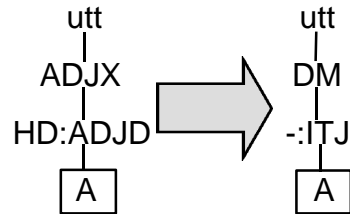
Die Transformationen überführen einen Syntaxbaum in einen anderen Syntaxbaum



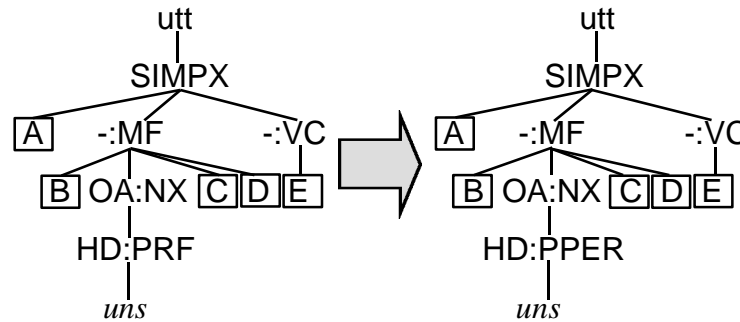
Typisch sind kleine Modifikationen

Typische Transformationen führen intuitiv sinnvolle Verbesserungen durch

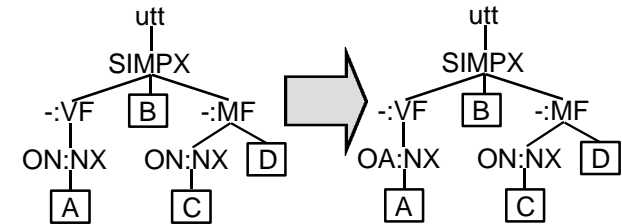
richtig, schön, gebongt



lassen Sie uns das so vereinbaren

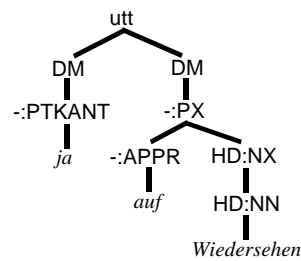
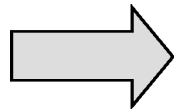
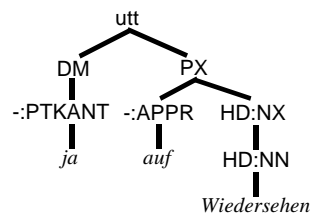


das finde ich gut, das denke ich auch

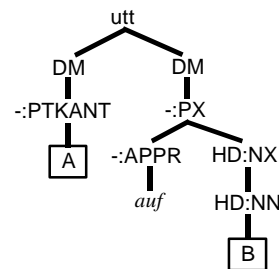
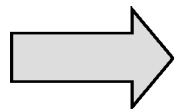
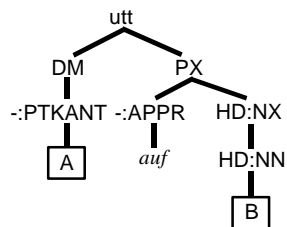


Verschlechterungen sind auch möglich, aber selten

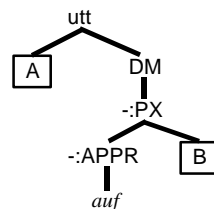
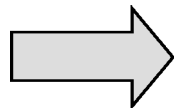
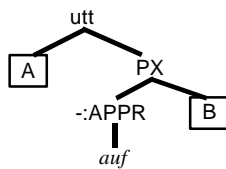
Lernziel ist die allgemeinste Abbildung eines falschen Parserbaums in ein Datum der Baumbank



trivial



zu speziell

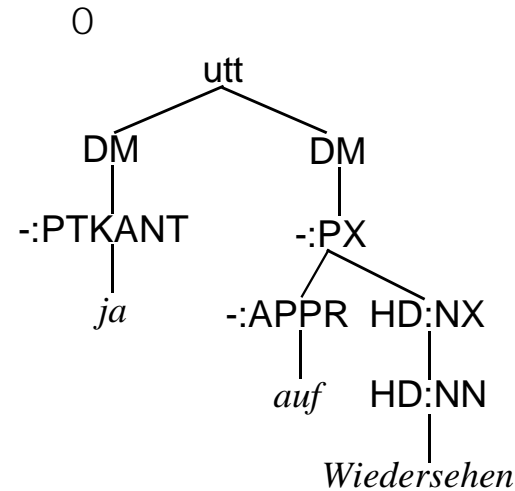
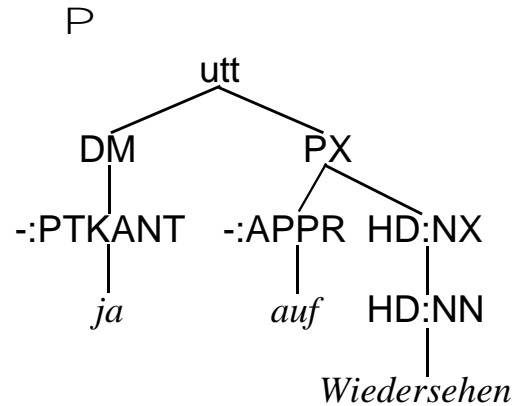


allgemein

Der Lernalgorithmus ist einfach aber effektiv

1. Finde die Menge G aller Gemeinsamkeiten zwischen Quellbaum P und Zielbaum O
2. Bilde die Menge T aller möglichen Transformationen. Eine Transformation t entsteht durch die Ersetzung (θ_i) einer oder mehrerer Elemente aus G durch Variablen in P und O
($t: \theta_i(P) \Rightarrow \theta_i(O)$)
3. Wähle die „optimale“ Transformation aus T aus

„Gemeinsamkeiten“ ist die Schnittmenge aller auftretenden Subterme



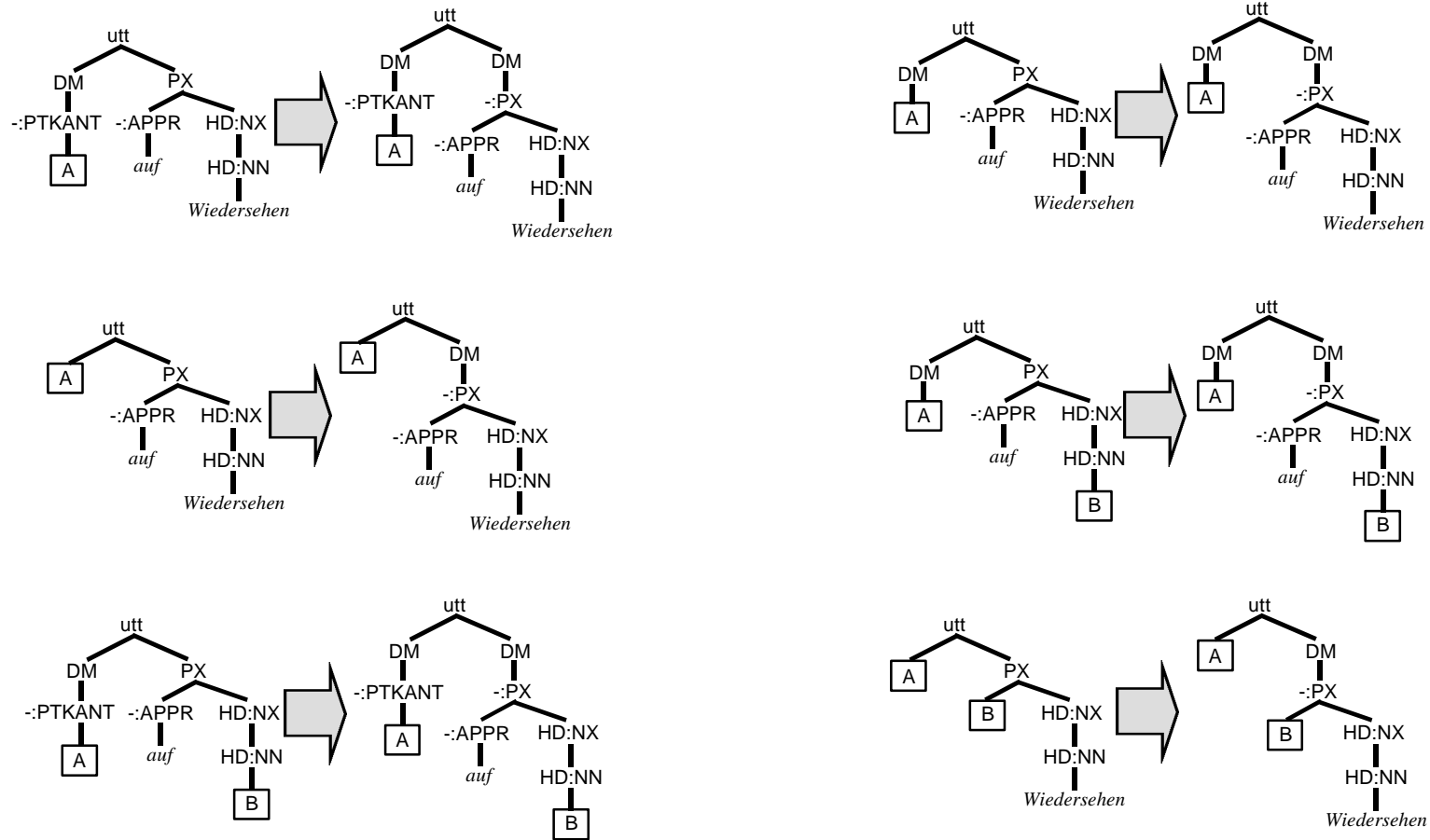
$[_{utt} [_{DM} [_{-:PTKANT} ja]] [_{PX} [_{-:APPR} auf] [_{HD:NX} [_{HD:NN} Wiedersehen]]]]]$

$[_{utt} [_{DM} [_{-:PTKANT} ja]] [_{DM} [_{-:PX} [_{-:APPR} auf] [_{HD:NX} [_{HD:NN} Wiedersehen]]]]]]$

$$G = \text{subterms}(P) \cap \text{subterms}(O)$$

$$G = \{ ja, [_{-:PTKANT} ja], [_{DM} [_{-:PTKANT} ja]], auf, [_{-:APPR} auf], Wiedersehen, [_{HD:NN} Wiedersehen], [_{HD:NX} [_{HD:NN} Wiedersehen]] \}$$

Das Ersetzen der Gemeinsamkeiten generiert eine Vielzahl möglicher Transformationen



und einige weitere...

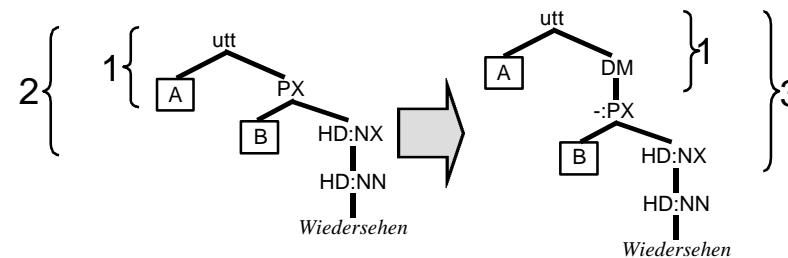
Die Auswahl der „optimalen“ Transformation ist ein mehrdimensionales Problem

- ✘ Anzahl akzeptierter Fehler auf dem Trainingsmaterial
(wieviele korrekte Bäume werden fälschlicherweise transformiert?)
- ✘ Abdeckung des Korpus
(wieviele „Treffer“ im Trainingsmaterial sind notwendig?)
- ✘ Verallgemeinerungsgrad
(wie gut generalisiert die Transformationsregel?)

Gute Resultate auf *Verbmobil*-Daten ergab folgende Parametrisierung:

- keine tolerierten Fehler
- mind. ein Beleg im Korpus
- “minimales mittleres Abstandskriterium“

Das „minimale mittlere Abstandskriterium“ misst den gemittelten Variablenabstand vom Wurzelknoten innerhalb der Transformationsregel



Mittlerer Abstand der Transformationsvariablen vom Wurzelknoten:

$$[(1+2) + (1+3)] / 4$$

Sind mehrere Transformationen „optimal“, wird eine zufällig gewählt

Zusammenfassung Transformationslernverfahren

Gegeben seien Quellbaum P und Zielbaum O in Termdarstellung.

1. $G = \text{subterms}(P) \cup \text{subterms}(O)$
2. Sei $S = \{s_1, s_2, \dots, s_m\}$ die Menge aller m Teilmengen von G , mit der Bedingung, dass $\exists a \in s_i, \exists b \in s_i$ mit $a @ b$ gelte $a \text{hsubterms}(b)$ und $b \text{hsubterms}(a)$.
3. Bilde $T = \{ \theta_1(P) \mapsto \theta_1(O), \theta_2(P) \mapsto \theta_2(O), \dots, \theta_m(P) \mapsto \theta_m(O) \}$, die Menge aller möglichen m Transformationen. θ_i sei die Ersetzung der Elemente von s_i durch logische Variablen.
4. Wähle $t_{\text{final}} \in T$, welche das Optimalitätskriterium erfüllt. Gibt es mehrere solche Elemente aus T , wähle eine dieser Abbildungen zufällig aus.

„Lerne diejenige Abbildung eines Baumes P auf einen Baum O , die auf dem Trainingsmaterial keine Fehler generiert, die maximale Abdeckung im Korpus erreicht und den geringsten mittleren Abstand der enthaltenen Variablen zum Wurzelknoten in Quell- und Zielstruktur besitzt.“

Für andere Korpora können ganz andere Kriterien optimal sein!

Erklärung der verwendeten Evaluationsmaße

Exact Match (EM)

gibt an, in wievielen Prozent der geparsten Testäußerungen die exakt richtige Struktur zugewiesen wurde.

Labeled Recall (LR)

$$\frac{\text{Anzahl der korrekt generierten nicht-terminalen Konstituenten}}{\text{Anzahl nicht-terminaler Konstituenten in der korrekten Analyse}}$$

Labeled Precision (LP)

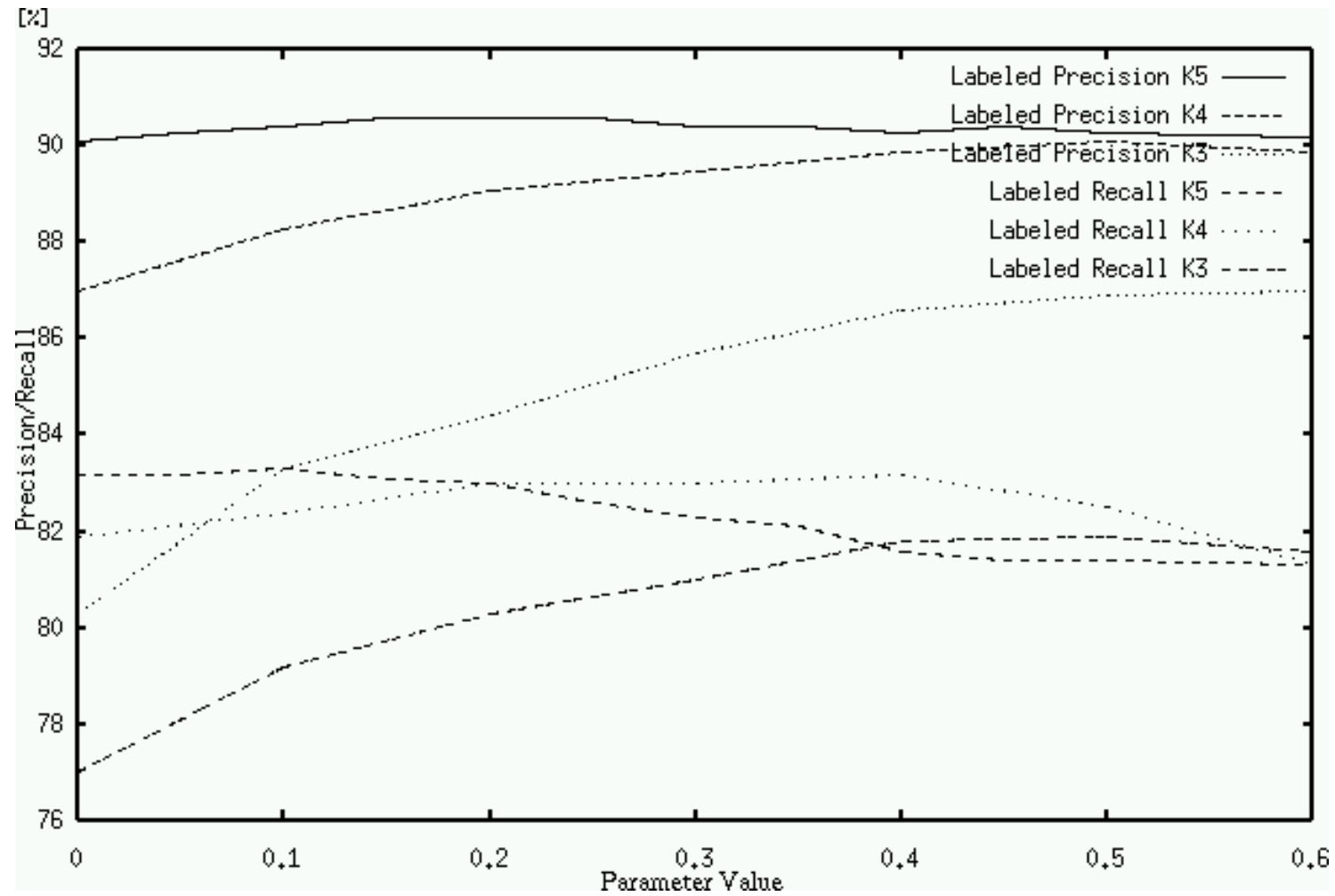
$$\frac{\text{Anzahl der korrekt generierten nicht-terminalen Konstituenten}}{\text{Anzahl aller generierten nicht-terminalen Konstituenten}}$$

Ergebnisse des stochastischen Parsers

	<i>Deutsch</i>	<i>Englisch</i>	<i>Japanisch</i>
Trainingsbäume	19.750	17.793	3.218
Testäußerungen	1.000	1.000	300
davon richtig geparsed (<i>Exact Match</i>)	46,3%	55,4%	67,7%
“ mit Fehler “	50,3%	39,3%	21,3%
“ nicht “	3,4%	5,3%	11,0%
Anzahl Grammatikregeln	988	2.205	932
Anzahl LALR-Tabelleneinträge	83.653	261.027	61.573
Labeled Precision	90,2%	90,6%	84,9%
Labeled Recall gesamt	83,5%	78,5%	63,1%
Labeled Recall auf erfolgreichen Analysen	91,0%	90,9%	86,3%
Durchschn. Verarbeitungszeit 166 MHz Sparc	425 ms.	430 ms.	30 ms.

Diese Ergebnisse wurden ohne Baumtransformationen gemessen

Ergebnisse mit verschiedenen Kontexten



Die Baumtransformationen steigern *Exact Match* um 6-8% absolut

	<i>Deutsch</i>	<i>Englisch</i>
<i>Exact Match</i> (ohne BT)	46,3%	55,4%
mit Fehlern geparsed (ohne BT)	50,3%	39,3%
nicht geparsed	3,4%	5,3%
<i>Exact Match</i> (mit BT)	53,8%	61,2%
Bäume mit Fehlern (mit BT)	42,8%	33,5%
<i>Labeled Precision</i> (ohne BT)	90,2%	90,6%
<i>Labeled Precision</i> (mit BT)	90,8%	91,4%
<i>Labeled Recall</i> ges. (ohne BT)	83,5%	78,5%
<i>Labeled Recall</i> ges. (mit BT)	84,0%	79,2%
<i>Labeled Recall</i> erfolgr. (ohne BT)	91,0%	90,9%
<i>Labeled Recall</i> erfolgr. (mit BT)	91,6%	91,7%

Für das Japanische konnte keine Verbesserungen erzielt werden.

Ergebnisse der Analyse spontansprachlicher Wortgraphen

Daten: Erkenner RWTH, *Verbmobil*-Akustik-Evaluation Herbst 99,
375 deutsche Turns, Mensch-Mensch-Kommunikation
Wortfehlerrate 26,7%, Satzfehlerrate 72,6%

Analyse: 240 von 375 Turns konnten analysiert werden (64%)

Semantik: 490 Terme für 237 Turn-Analysen, für 3 keine Semantik

Transfer: im Dezember 99 konnten 99% aller Semantikterme durch
das Transfermodul übersetzt werden

⇒ etwa 62% der Wortgraphen wurden in eine zielsprachl. Semantik übersetzt

Performanzvergleich mit anderen Analysemodulen auf derselben Aufgabe

Ergebnisse des „1000-Turn-Experiments“ im November 1999

Daten: prosodisch annotierte Transliterationen von 1576 Turns, davon 1081 in deutscher, 495 in englischer Sprache

Experiment: komplette Analyse und Übersetzung der Daten

Ergebnis (durch Menschen ausgewertet):

	<i>Stoch. Parser</i>	<i>HPSG</i>	<i>Chunkparser</i>
Turnkorrektheit auf 1576 Turns	55,6%	54,9%	57,7%
Turnkorrektheit Deutsch	56,9%	61,4%	60,0%
Turnkorrektheit Englisch	53,3%	40,6%	52,7%
Turnkorrektheit auf ungesehenen	44,4%	43,2%	34,6%
Segmentkorrektheit auf allen Turns	63,7%	55,4%	60,6%
Segmentkorrektheit auf ungesehenen	47,0%	35,6%	24,1%



Performanzvergleich mit anderen Analysekomponenten (II)

Ergebnisse der End-zu-End-Evaluation im Dezember 1999

Experiment: reale deutsch-englische End-zu-End-Dialogsituation
5 Dialoge, aufgenommen von Universität Hamburg

End-zu-End-Übersetzungsqualität (durch Menschen ausgewertet):

Analyse durch...	<i>gut</i>	<i>mittelmäßig</i>	<i>schlecht</i>	<i>nicht übersetzt</i>
Stoch. Parser	26,0%	17,8%	37,8%	18,4%
HPSG-Parser	23,8%	20,0%	14,6%	41,6%
Chunkparser	15,1%	22,2%	54,0%	8,7%

Beispielhafte Übersetzungen aus Analysen des vorgestellten Parsers

Übersetzungsstrang Daimler_16kHz→Parser→Semantik→Transfer→Generierung

guten Tag

hello

wir müssen noch einen Termin ausmachen

we still must schedule a meeting

wann würde es Ihnen denn passen?

when would that work for you?

am Montag geht es bei mir nicht

that doesn't work for me on monday

der dreizehnte ist leider schlecht

the thirteenth is unfortunately bad

kennen Sie ein Hotel in München?

you know a hotel in Munich?

abends können wir ins Theater gehen

we can leave into the theater in the evening

dann bin ich nämlich in Berlin

because I'm in Berlin

gut

fine

ich könnte dann wieder ab dem 15.

that is possible for me since the fifteenth again

am Donnerstag bin ich in Hannover

I will be in Hanover on thursday

aber da kann man nix machen

but one can make nothing there

Zusammenfassung der Ergebnisse

Im Rahmen der vorgestellten Arbeit wurde

- ✘ der erste stoch. LR-Parser für drei Sprachen geschaffen, der große Kontexte während der Analyse berücksichtigt und auf geparsten spontansprachlichen Äußerungen etwa 90% Labeled Precision/Recall erreicht,
- ✘ wurde demonstriert, dass Parser dieser Bauart geeignet sind, in einem realen Übersetzungssystem eingesetzt zu werden,
- ✘ bewiesen, dass der entwickelte Parser mit anderen Ansätzen auf derselben Aufgabe durchaus konkurrieren kann (bei deutlich niedrigerem Entwicklungsaufwand),
- ✘ gezeigt, dass sich die Leistung eines stochastischen Syntaxanalyseansatzes deutlich mit einem symbolischen Nachbearbeitungsschritt verbessern lässt.

Ausblick

Das vorgestellte Verfahren bietet mehrere Ansatzpunkte für Verbesserungen:

1. Algorithmische Optimierung des Suchprozesses,
(*Structure Sharing, Optimierung A**)
2. Vergrößerung der symbolischen Restriktionen beim Parsing,
(*manuell erstelltes Lexikon, Lernen symbolischer Restriktionen [Osb94]*)
3. Arbeiten zur Optimierung des Kontextes und
(*Theoretische Arbeiten zur Auswahl der Kontexte, datenabhängige Kontexte*)
4. Verfeinerung des regelbasierten Baumtransformationsansatzes.
(*Abbildung von Teilstrukturen statt kompletter Bäume*)

Literatur

- [Bri93] Brill, E. *A Corpus-Based Approach To Language Learning*. Dissertation, University of Pennsylvania, 1993.
- [BC93] Briscoe, T., Carrol, J. *Generalized Probabilistic LR Parsing of Natural Language (Corpora) with Unification-Based Grammars*. In: *Computational Linguistics*, Vol. 19, No. 1, 1993.
- [IST97] Inui, K. et al. *A New Formalization of Probabilistic GLR Parsing*. In: *Proceedings of the International Workshop on Parsing Technologies*, 1997.
- [Lav96] Lavie, A. *GLR*: A Robust Grammar-Focused Parser for Spontaneously Spoken Language*. Dissertation, Carnegie Mellon Universität, 1996.
- [Osb94] Osborne, M. *Learning Unification Based Natural Language Grammars*. Dissertation, Universität York, Großbritannien, 1994.
- [Sch94] Schmid, L. *Parsing Word Graphs Using a Linguistic Grammar and a Statistical Language Model*. In: *Proceedings of the ICASSP '94*, Adelaide, 1994.
- [Web94] Weber, H. *LR-inkrementelles, probabilistisches Chartparsing von Worthypothesengraphen mit Unifikationsgrammatiken: Eine enge Kopplung von Suche und Analyse*. Dissertation, Universität Hamburg, 1994.
- [WW91] Wright, J.H., Wirygley E.N. *GLR Parsing with Probability*. In: Tomita, M. (Hrsg.) *Generalised LR Parsing*. Kluwer, Boston, 1991.
- [Zel95] Zelle, J. M. *Using Inductive Logic Programming to Automate the Construction of Natural Language Parsers*. Dissertation, University of Texas, Austin, 1995.