

Hybride Methoden zur automatischen Analyse natürlicher Sprache

Kilian A. Foth

Natural Language Systems Group
MIN-Fakultät, Department of Informatics
Universität Hamburg
foth@informatik.uni-hamburg.de

Abstract: Eine handgeschriebener Parser des Deutschen wird um statistische Methoden ergänzt, die dessen axiomatische und operative Semantik verbessern. Das Gesamtsystem weist dabei höhere Präzision als jede Einzelkomponente auf und definiert eine neue Bestmarke für automatische deutsche Parser.

1 Einführung

Seit der Erfindung moderner Digitalcomputer hat es Versuche gegeben, sie anstelle von speziellen Programmiersprachen unsere eigene, natürliche Sprache zu lehren. Der Vorteil eines solchen Systems läge auf der Hand: anstelle von speziellen und un-intuitiven Anweisungen wie etwa

```
[QUERY (WH employeel
        (AND (EMPLOYEE employeel)
              (EMPLOYEES-COMPANY-OF employeel SRI))
        (SOME child2
          (CHILD child2)
          (AND (CHILD-OF employeel child2)
               ((*MORE* OLD) child2 (YEAR 15)))))]
```

könnte etwa eine Datenbankabfrage mit der unmittelbar verständlichen Formulierung “Welche Angestellten von SRI haben Kinder über 15 Jahren?”

gestellt werden¹. Wie die meisten Aufgaben der Künstlichen Intelligenz ist dieses Ziel jedoch bis heute unerreicht geblieben.

In den Anfangszeiten der Informatik wurde allgemein angenommen, daß menschliche Sprache als grundsätzlich regelbasiertes Phänomen ideal zur Verarbeitung durch Rechenmaschinen geeignet sein würde. Gab es mit Lexika und Grammatiken nicht umfassende

¹Dieses Beispiel ist [GHH⁺82] entnommen.

Regelwerke, die jede Einzelheit unserer Sprache genau beschreiben? Ein Computerprogramm etwa zur Übersetzung zwischen zwei bestimmten Sprachen wäre damit im wesentlichen durch ein maschinenlesbares Lexikon und eine leicht erstellbare Sammlung von Abbildungsregeln zu realisieren. Tatsächlich erwies sich der notwendige Aufwand aber als unvergleichlich größer; noch 50 Jahre nach den ersten Versuchen hat die vollautomatische Übersetzung Mühe, ein Ergebnis zu liefern, das auch nur als Ausgangsmaterial für einen menschlichen Experten ökonomisch sinnvoll ist.

Zu diesen Schwierigkeiten tragen verschiedene Eigenheiten unserer Sprache bei, die uns erst durch die ersten computerlinguistischen Versuche überhaupt bewußt wurden. Zum einen ist sie kombinatorisch *mehrdeutig*, so daß ein exponentiell ansteigender Verarbeitungsaufwand für den Rechner entsteht: selbst eindeutig erscheinende Sätze können nach den Regeln einer formalen Grammatik tausende oder gar Millionen verschiedene Interpretationen besitzen. Während der Mensch durch Hintergrundwissen und Heuristik die korrekte Struktur erschließt, ohne sich der vielen sinnlosen Alternativen überhaupt bewußt zu werden, zählt der Rechner diese lediglich auf und erzeugt damit ein nutzloses Überangebot an Lösungen. Zum anderen ist der Mensch in der Lage, Äußerungen *robust* zu verstehen, selbst wenn sie sogar die weitreichenden Freiheiten der Grammatik verletzen. Jede realistische Konversation zwischen Menschen nutzt diese Freiheiten weidlich aus; paradoxerweise ist Regelkonformität in spontaner Sprache die Ausnahme und nicht die Regel.

Diese beiden Probleme stellen ein schweres Dilemma dar. Eine völlige Erfassung aller Varianten des menschlichen Ausdrucks durch Regeln ist extrem aufwendig, und eine künstliche Vereinfachung des Untersuchungsgegenstandes läuft Gefahr, eben nur eine Computersprache statt natürlicher Sprache abzubilden. Der Versuch, die Abdeckung eines Systems durch zusätzliche Regeln, Varianten, Auslassungsmöglichkeiten etc. zu erweitern, führt aber wiederum dazu, daß die Mehrdeutigkeit von bereits abgedeckten Sätzen für den Rechner noch weiter zunimmt und das ursprüngliche Problem damit verschärft.

Die Computerlinguistik hat in Reaktion auf diese Schwierigkeiten ihre frühen Ansprüche weitgehend ad acta gelegt und erforscht vornehmlich Modelle für hypothetische einzelne Komponenten des Sprachverstehensprozesses wie akustische Worterkennung, Syntaxberechnung, Stimmungserkennung etc. Diese Arbeit widmet sich der Syntaxanalyse des Deutschen im Rahmen der *gewichteten Constraint-Dependenz-Grammatik (WCDG)*. Es wird zum einen ein umfassendes Modell des Deutschen vorgestellt, das durch die Konzeption der Grammatik als Sammlung verletzbarer Constraints anstelle harter Regeln eine vollständige Abdeckung menschlicher Äußerungen anstrebt. Dabei erlaubt die Gewichtung von Constraints eine Priorisierung der unweigerlich entstehenden Alternativen untereinander, so daß im Prinzip *jeder* Äußerung eine eindeutige präferierte Lesart zugeordnet werden kann. Dieses Modell erzielt bereits eine konkurrenzfähige Präzision in der Syntaxanalyse. Zum anderen wird neben dem axiomatischen Ansatz die Möglichkeit untersucht, auch empirisches Wissen für den Parser nutzbar zu machen, um Schwachstellen der Grammatik gezielt zu beheben, insbesondere in der Wortartbestimmung und dem Phänomen der lexikalisierten Präferenzen.

2 Parsing mit axiomatischem und empirischem Wissen

Mit dem Aufkommen immer schnellerer Prozessoren, größerer Korpora und höherer Bandbreiten ist die Nutzung von *empirischem* Wissen wieder stärker ins Blickfeld der Computerlinguistik gerückt. Anstelle von *axiomatischen* Systemen, die von Experten manuell Regel für Regel aufgeschrieben werden, wird dabei die tatsächliche massenhafte Verwendung von Sprache gemessen, aufbereitet und zur Vorhersage von Lösungen neu auftretender Probleme verwendet. Etwa bei der Sprachsignalerkennung, der maschinellen Übersetzung und dem Parsing haben empirische Modelle die Computerlinguistik grundlegend revolutioniert. Die Vorzüge liegen auf der Hand: ein sorgfältig ausgewähltes Korpus spiegelt die tatsächlich relevanten Phänomene getreu wieder, ohne daß der Systemautor sie ausdrücklich vorhersehen muß; Annotatoren sind billiger als Grammatiker; ein automatisch erzeugtes Modell läßt sich leichter auf andere Sprachen übertragen als ein maßgeschneiderter einsprachiger Regelsatz; Korpusfrequenzen lassen sich zur Disambiguierung zwischen unterschiedlich wahrscheinlichen Analysen einsetzen.

Das Versprechen von robuster und disambiguierender automatischer Analyse hat freilich seinerseits grundsätzliche Grenzen. Maschinenlesbarer Text ist zwar fast unbegrenzt verfügbar, nicht aber Analysen davon; das Bereitstellen von großen, balancierten und gleichzeitig einheitlichen Korpora erfordert seinerseits großen Aufwand und Disziplin. Oft sind umfassende Annotationsrichtlinien erforderlich, die ihrerseits axiomatischen Grammatiken ähneln. Automatische Optimierung ist meist nur unter vereinfachenden Annahmen traktabel, die linguistisch offensichtlich nicht gerechtfertigt sind. Schließlich kann man offensichtliche Fehler in einem automatisch generierten Modell nicht beheben, ohne dadurch das Gesamtverhalten unvorhersehbar zu beeinflussen. Diese Schwächen des empirischen Ansatzes sind gerade die Stärken des axiomatischen. Obwohl also derzeit empirische Methoden der Stand der Kunst sind, könnte die Kombination von beiden insgesamt noch bessere Ergebnisse erzielen.

Der hier angewendete Parsing-System WCDG gehört grundsätzlich zu den axiomatischen Formalismen. Die Oberflächensyntax von Sätzen wird durch *Dependenzbäume* modelliert, also gerichtete azyklische Graphen, deren Knoten die Wortformen bilden und deren Kanten jeweils eine funktionale Abhängigkeit zwischen zwei Worten abbilden. Zur Erhöhung der Ausdruckskraft tragen die Kanten Beschriftungen, die etwa die Unterscheidung zwischen Subjekten und Objekten oder zwischen freien und idiomatischen Präpositionalkonstruktionen erlauben; die hier verwendete Grammatik des Deutschen unterscheidet 35 solcher Label. Die Wurzel des Dependenzbaumes bildet ein spezieller Wurzelknoten, der keinem Wort entspricht; wenn gewünscht, kann ein Dependenzbaum ohne weiteres auch mehrere solche Wurzelknoten enthalten, um fragmentarische oder elliptische Konstruktionen zu modellieren.

Sprachliches Wissen wird in Form von *Constraints* notiert, aussagenlogischen Formeln, die die Struktur wohlgeformter Sätze beschreiben. Etwa 1000 Constraints wurden formuliert, um eine umfassende Grammatik des Neuhochdeutschen zu konstruieren. Sie reichen in der Komplexität von einfachen Agreement-Forderungen über die beiden Wordformen etwa an Subjekt- und Objektkanten bis zu komplexen strukturellen Bedingungen über ganze Bäume, die z.B. festlegen, welche deutschen Konstruktionen die Projektivitätsei-

genschaft einhalten müssen und welche nicht. Die Formulierung einer Grammatik durch deklarative Constraints über Dependenzbäume im Gegensatz zu generativen Regeln ist besonders von Vorteil für Sprachen mit relativ freier Wortstellung, da Varianten der linearen Anordnung nicht eigens erlaubt, sondern lediglich nicht eingeschränkt werden müssen.

Da die Mehrzahl der gefundenen Regeln keineswegs den Charakter von lizensierenden Bedingungen hat, sondern vielmehr auf real vorkommende Äußerungen nur tendentiell zutreffen, ist die Mehrzahl der Constraints durch *Gewichtung* als verletzbar deklariert. Die gewünschte Analyse ist stets diejenige, die in der Summe die Constraints mit den geringsten Gewichten verletzt. So ist etwa die Hauptsatzreihenfolge Subjekt-Verb-Objekt im Deutschen wesentlich weniger starr als im traditionell untersuchten Englischen; das entsprechende Constraint ist daher lediglich als schwache Präferenz formuliert, so daß ein Parser in Ermangelung anderer Indikatoren diese Konstruktion annehmen wird, aber in markierten Sätzen nicht an der Analyse gehindert wird. Die Kongruenz von Kasus, Numerus und Genus ist als ein stärkeres Constraint anzusehen, aber da empirisch immerhin 0.4% aller Sätze in Nachrichtenartikeln gegen sie verstoßen, sind auch diese Constraints nur als stärkere Präferenzen formuliert. Lediglich grundlegende Bedingungen wie etwa die, daß Subjektbeziehungen nur zwischen Verb- und Nominal-Material angenommen werden sollten, werden als unverletzlich angesehen. Dadurch ist eine Robustheit gleichermaßen gegen *extragrammatische* (akzeptable, aber nicht vorhergesehene) Phänomene gegeben wie gegen tatsächlich *ungrammatischen* Input (deutlich als solche erkennbare Regelverstöße, die jedoch die Verständlichkeit nicht behindern müssen). Ein Parser hat damit die Möglichkeit, einerseits eine Selbsteinschätzung abzugeben, wie sicher er sich seiner Analyse ist; gleichzeitig kann er z.B. im Sprachunterricht erkannte Fehler ausführlich lokalisieren, erklären und sogar offensichtliche Korrekturen berechnen.

Die hohe Ausdrucksmächtigkeit von WCDG - durch seine aussagenlogischen Formeln lassen sich beliebige formale Sprachen beschreiben - hat freilich ihren Preis: nicht nur die Ermittlung der numerisch optimalen Struktur zu einem Input, sondern schon das Wortproblem ist \mathcal{NP} -vollständig und damit exponentiell aufwendig zu lösen. Obwohl die Anwendung großer Scharen unabhängiger Constraints im Prinzip sehr gut zur Verarbeitung auf parallelen Rechnerarchitekturen geeignet ist, ist damit eine exakte Implementation der Maximumsuche praktisch schwer zu gewährleisten. In der Praxis hat sich jedoch herausgestellt, daß transformative Anytime-Algorithmen, die ausgehend von einer suboptimalen Gesamtanalyse das numerische Optimum heuristisch approximieren, das theoretische Optimum in guter Näherung berechnen können, so daß der Präzisionsverlust durch Suchfehler gering bleibt. Als eine gute Lösung hat sich dabei erwiesen, jeweils diejenigen Dependenz einer Analyse systematisch abzuändern, die derzeit die stärksten Constraint verletzen[Fot99]. So läßt sich bereits die einfache WCDG zur Entwicklung von Parsern mit konkurrenzfähiger Präzision verwenden.

3 Hybridisierung von WCDG

Trotz dieser Qualitäten ist die größte Hürde für den erfolgreichen Einsatz von WCDG der übermäßige Aufwand sowohl bei der Entwicklung als auch im Einsatz. Hier versprechen

empirische, insbesondere statistische Methoden Abhilfe: ein selbsttätig lernendes System muß nicht mehr ausdrücklich mit allen Einzelheiten einer Anwendungsdomäne programmiert werden, und einfache statistische Wahrscheinlichkeitsmaße können schnell partielle oder approximative Lösungen zu sprachlichen Problemen liefern, die trotz geringerer Präzision einem Parser von Nutzen sein können. Es wurden insgesamt fünf solche Methoden untersucht, die sich allesamt als wertvoll zur Unterstützung eines WCDG-Parsers herausstellen. Stellvertretend werden hier drei solche Komponenten erläutert: Die *statistische Kategoriebestimmung*, lexikalisierte *Präpositionalanbindung* sowie ein *Orakel-Parser*.

3.1 POS-Tagging

Das *part-of-speech tagging* von Texten weist jeder Wortform die im Kontext wahrscheinlichste Wortartenkategorie zu. Diese Aufgabe ist bereits durch relativ einfache n -Gramm-Modelle mit hoher Zuverlässigkeit ($> 95\%$) lösbar, wäre aber in WCDG-Constraints nur umständlich zu implementieren. Stattdessen wurde ein existierender Tagger als Präprozessor an WCDG angebunden, dessen Voraussagen durch ein spezielles dynamisches Constraint als weitere Präferenzen in die Optimierung einfließen.

Im vorliegenden Modell des Deutschen erweist sich dabei insbesondere die Reduzierung des Suchraumes um gänzlich abwegige Kategorievarianten als sehr hilfreich. Aufgrund der beabsichtigten vollständigen Abdeckung auch offener Wortklassen erlaubt die Grammatik nämlich, grundsätzlich jede morphologisch passende Wortform als Nomen, Verb etc. anzusehen, was jedoch den Parsingaufwand noch einmal stark erhöht. Bei der Filterung dieser Möglichkeiten durch einen Präprozessor werden falsche Varianten sehr zuverlässig ausgeschlossen und gleichzeitig die Abdeckung für unbekannte Worte erhalten.

Insgesamt bewirkt allein die Anbindung des Taggers beim Parsen des NEGRA-Testsets [BSK97] eine Anbindungsgenauigkeit von annähernd 90%; die Strukturfehlerrate sinkt damit um fast zwei Drittel gegenüber der stark übergenerierenden Basisgrammatik. Der Ausschluß der unwahrscheinlichsten Varianten erlaubt also, die eingesparte Zeit zur besseren Optimierung unter den wahrscheinlicheren zu verwenden, und die Genauigkeit des Gesamtsystems steigt, obwohl das Hilfsprogramm seinerseits noch eine erhebliche Fehlerquelle darstellt. Fehler des Taggers selbst werden einerseits durch die gewichtete, also nicht-absolute Integration seines Outputs kompensiert, zum anderen durch spezielle handgeschriebene Regeln, die charakteristische, durch den n -Gramm-Horizont des Taggers verursachte Fehler korrigieren. Auch die Kategoriebestimmung selbst profitiert also bereits von einer hybriden Strategie, die die Stärken beider Paradigmen ausnutzt.

3.2 PP-Attachment

Die größte verbleibende Fehlerquelle im Output des so erweiterten WCDG-Systems ist die strukturelle Anbindung von Präpositionen. Diese sind einerseits sehr zahlreich, andererseits notorisch schwer zu disambiguieren. Im untersuchten Testset sind über 3% aller Wor-

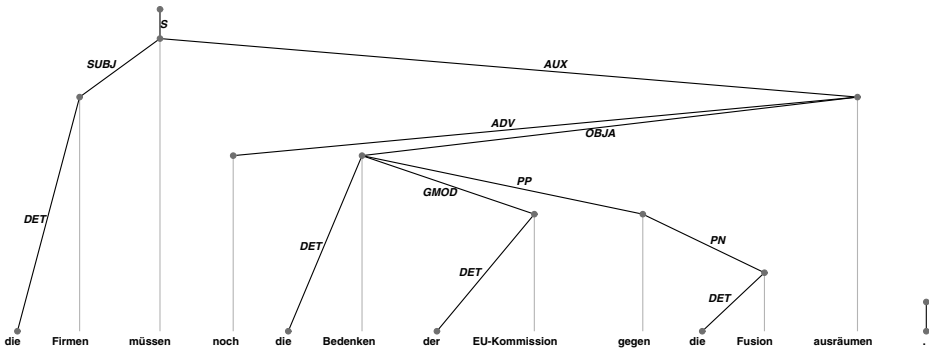


Abbildung 1: Ein Dependenzbaum mit idiomatischer PP-Anbindung.

te falsch angebundene Präpositionen; die Präpositionalbeziehung hat damit die geringste Präzision aller gängigen Dependenztypen. Ein wesentlicher Grund für diese Schwachstelle ist die fehlende *Lexikalisierung* der vorliegenden Grammatik. So sind zwar allgemeine Präferenzen bei der Präpositionalkonstruktion leicht zu formulieren (eher kurze als lange Anbindung, eher Verb- als Nominalmodifikation), diese sind jedoch oft nebensächlich verglichen mit idiomatischen Beziehungen, die die spezifischen Worte eines Satzes zueinander aufweisen (vgl. Abbildung 1).

Obwohl z.B. die Kombination ‘Bedenken gegen’ eine offensichtliche Präferenz aufweist, enthält die Basisgrammatik keine entsprechende Regel, die die allgemeineren Präferenzen überstimmen könnte. Das manuelle Auffinden und Notieren solcher Kombinationen wäre äußerst aufwendig und fehleranfällig. Hier bietet sich wiederum eine empirische Lösung an. Es wurden zum einen etwa 100.000 im Rahmen der Arbeit entstandene, manuell revidierte Syntaxbäume und zum anderen 295.000.000 Worte an Rohtext auf Kollokationen zwischen Präpositionen und Regenten bzw. benachbarten Nomen und Verben untersucht. Anhand der Auftretsfrequenzen von Einzelworten und Wortpaaren wurde das offensichtliche Korrelationsmaß berechnet; selbst ein unüberwachter Algorithmus kann dadurch lernen, daß die Kollokation ‘Bedenken gegen’ eine Größenordnung wahrscheinlicher ist als alle Alternativen in diesem Satz.

Mannigfache Komplikationen wie etwa Verzerrungen aufgrund der systematischen Beziehungen zwischen mehr als zwei Worten, der sinnändernde Einfluß abtrennbarer Verbpartikel, sowie die Beschränkung des verwendeten Lernalgorithmus, der z.B. nicht das Präpositionalnomen als Feature auswertet, können in einzelnen Fällen die richtige Parsingentscheidung zunichtemachen. Dennoch kann diese zusätzliche Entscheidungshilfe insgesamt die strukturelle Genauigkeit des Parsers um 1.5 Prozentpunkte erhöhen; es wird also ungefähr die Hälfte des theoretisch möglichen Nutzens einer Präpositions-Prädiktion tatsächlich erreicht. Dies ist um so bemerkenswerter, als die einfache Prädiktionskomponente selbst deutlich hinter den Ergebnissen dedizierter Lösungen zurückbleibt, also offensichtlich gar nicht alle Kollokationen aufdeckt, die die Trainingsdaten definieren. Eine weitere Verbesserung der Ergebnisse durch Einbindung einer Komponente, die dem Stand der

Kunst entspricht, erscheint daher möglich.

3.3 Orakel-Parser

Als letzte externe Komponente zur Unterstützung von WCDG wurde ein statistischer Dependenzparser verwendet. Diese Maßnahme erscheint zunächst widersinnig. Wie kann ein externer Prädiktor, der genau dieselbe Aufgabe löst wie das Hauptprogramm, der Analyse von Nutzen sein, ohne dieses an Leistung zu übertreffen? Es hat sich jedoch insbesondere auf dem Gebiet des automatischen Parsings gezeigt, daß die Zusammenschaltung mehrerer unabhängiger Systeme die Genauigkeit über die beste Einzelleistung hinaus steigern kann und daß dabei unter Umständen die Diversität der Ergebnisse sogar wichtiger ist als individuelle Exzellenz [ZZ05].

Im Fall von WCDG ist es insbesondere das unterschiedliche Laufzeitverhalten verschiedener Parser, das eine Hybridlösung sinnvoll macht. Constraintgrammatiken können auch komplexe Sprachmodelle sehr präzise formalisieren, stellen dafür aber sehr schwere Optimierungsprobleme auf. Die Schwierigkeit erhöht sich noch beträchtlich, wenn wie in dieser Arbeit eine sehr robuste Abdeckung auch ungrammatischer Sprache gefordert wird. Insbesondere werden selbst bei der Analyse von einfacher, voll grammatischer Sprache sehr viele Alternativen zumindest zeitweise erwogen, deren Aufzählung sehr zeitaufwendig ist, selbst wenn sie jeweils nach kurzer Abwägung verworfen werden. Insbesondere bei langen Äußerungen kann es dazu kommen, daß die Suchheuristiken aufgrund von Zeitmangel das numerische Optimum verfehlen, wo ein einfacheres Modell womöglich direkter zur offensichtlichen Lösung vorgestoßen wäre, d.h. die breite Abdeckung der Grammatik erweist sich als Hindernis für die Analyse in genau denjenigen Fällen, in denen sie nicht ausgenutzt wird. Da ein ungrammatischer Satz für den Computer aber eben erst dadurch zu erkennen ist, daß die Suche nach einer hoch bewerteten Lösung fehlschlägt, läßt sich dieses Problem grundsätzlich nicht ausräumen. Ein von Natur aus schneller, d.h. in Linearzeit arbeitender Parser kann hier nützlich sein, um sicherzustellen, daß die oberflächlich plausibelste zumindest als erste von vielen geprüft wird statt an undefinierter Stelle.

Um diesen Effekt zu demonstrieren, wurde wiederum ein verhältnismäßig einfaches empirisches Modell als Prädiktor verwendet. In Anlehnung an [Niv03] wurde ein einfacher Shift-Reduce-Parser implementiert und auf einer Dependenzversion des NEGRA-Korpus trainiert. Ein solcher Parser verwendet einen Stack von Worten und kann jeweils zwischen dem aktuellen Eingabewort und der Stackspitze eine Dependenz postulieren oder aber den Stack auf- oder abbauen. Hierdurch kann jeder projektive Dependenzbaum mit Kantenlabels in höchstens $2n$ Schritten konstruiert werden. Offensichtlich hängt die Präzision eines solchen Parsers davon ab, wie zuverlässig er anhand dieses eingeschränkten Kontextes globale Anbindungsentscheidungen lokal vorhersagen kann.

Wie zuvor wurden zum Training lediglich so viele statistische Features verwendet, wie ohne größeren Aufwand im Arbeitsspeicher gehalten werden können. Die Anbindungspräzision dieses Parsers selbst bleibt daher erheblich hinter vergleichbaren Veröffentlichungen zurück und übersteigt nicht 85%. Er zeigt jedoch zum einen, daß offensichtlich

die ‘einfachen’ Fälle die große Mehrzahl aller Sätze bilden und die volle Maschinerie eines ‘tiefen’ Parsers tatsächlich nur in der Minderheit aller Fälle überhaupt zum Tragen kommt, und zum anderen, daß ein einfaches Modell tatsächlich besser geeignet ist, einfache Parsingprobleme zu lösen: ein direkter Vergleich deckt zahlreiche Fälle auf, in denen der Orakel-Parser einzelne Sätze genauer analysiert als WCDG, und auch eine erhebliche Anzahl vollständig korrekter Analysen. Die Vorhersagen des Prädiktors über Regenten und Label jedes Wortes wurden wiederum durch ein zusätzliches Constraint in WCDG selbst integriert. Hierdurch ist zu erhoffen, daß das Gesamtsystem in den vielen einfachen Fällen zuverlässig jeweils die plausibelste Variante zuerst prüft und im Idealfall gar keine Alternativen zu untersuchen braucht, da keine Constraintverletzungen auftreten, die durch Transformationen behoben werden müßten.

Dieses Experiment erbrachte drei bemerkenswerte Erkenntnisse. Erstens kann in der Tat die Präzision des Gesamtsystems bedeutend gesteigert werden, obwohl nur ein weiterer Parser hinzugefügt wurde, der überdies für sich allein WCDG deutlich unterlegen ist: von unter 90% steigt die Anbindungspräzision auf 91.5%. Untersuchungen an Einzelfällen legen nahe, daß hierbei tatsächlich im wesentlichen *Suchfehler* von WCDG ausgebessert wurden, also Fälle, in denen das Basissystem das theoretische Optimum verfehlte, und nicht *Modellierungsfehler* der Grammatik. Dies bestätigt die gute Eignung empirischer Modelle zur Erledigung der ‘einfachen’ Arbeit, so daß Rechenaufwand automatisch dort konzentriert werden kann, wo er tatsächlich gebraucht wird. Zweitens besteht trotz der Prädominanz der einfachen Fälle die sinnvollste Kombination der Systeme darin, die Vorhersage des Orakel-Parsers als eine sehr schwache Empfehlung anzusehen; der gefundene optimale Wert 0.95 definiert eine Präferenz, die schwächer ist als fast alle bestehenden Constraints der Grammatik. Offenbar ist dieser Wert geeignet, im Normalfall die plausible Lösung lokal zu bevorzugen, ohne daß aber in den unweigerlichen Fällen von falschen Prädiktionen den Parser allzusehr an einer Revision zu hindern. Drittens verändert sich durch die Hybridisierung das Zeitprofil von WCDG in vorteilhafter Weise. Wie bei allen Anytime-Algorithmen erbringt das transformative Parsing tendentiell bessere Ergebnisse, je mehr Rechenzeit ihm zugebilligt wird. Da die zusätzliche Heuristik des Shift-Reduce-Parsers in vielen Fällen dieselben Ergebnisse wie WCDG allein definiert, diese jedoch früher findet, läßt sich *dieselbe* durchschnittliche Qualität wesentlich schneller erzielen. Die durchschnittliche Beschleunigung kann dabei bis zu einer Größenordnung betragen.

3.4 Kombination von Prädiktoren

Von besonderem Interesse ist das Ergebnis bei der Kombination mehrerer Prädiktoren mit WCDG. Tabelle 1 zeigt Ergebnisse für die drei dargestellten sowie eine *Chunk-Parsing*-Komponente (CP) und einen *Supertagger* (ST). Mit einer geringfügigen Ausnahme verbessert sich das Parsingergebnis bei jeder Hinzunahme von Komponenten. Trotz der zum Teil erheblich fehlerhaften Hilfskomponenten findet also keine Fehlerakkumulation statt. Das Komplettsystem aus allen sechs Komponenten erreicht annähernd 91% Präzision für Anbindung *und* Label, das bislang beste Ergebnis für einen Parser für Deutsch mit breiter Abdeckung.

Experiment	Prädiktoren	Genauigkeit	
		Anbindung	Anb.+Label
1	-	72.6%	68.3%
2	POS	89.0%	87.1%
3	Hybrid-POS	89.3%	87.5%
4	POS+CP	89.8%	88.0%
5	POS+ST	91.9%	90.5%
6	POS+PP	90.6%	88.9%
7	POS+SR	91.5%	89.8%
8	POS+PP+SR	91.4%	89.6%
9	POS+PP+ST	92.0%	90.6%
10	POS+ST+SR	92.2%	90.7%
11	alle fünf	92.3%	90.9%

Tabelle 1: Parsingergebnisse mit und ohne Prädiktoren.

4 Zusammenfassung

Die Hybridisierung erweist sich in verschiedener Hinsicht nützlich bei der Lösung des äußerst schweren Parsing-Problems. *Filter*-Komponenten wie ein Kategorieerkenner können durch die weitgehende Einschränkung eines theoretisch fast unbegrenzten Suchraumes die großflächige Verarbeitung mit einer sehr robusten Grammatik überhaupt erst ermöglichen. Weiterhin können solche Regeln, die als Constraints nur mit sehr hohem Entwickleraufwand formuliert werden könnten, unter bestimmten Bedingungen gut durch empirische Prädiktoren ersetzt werden. Zum Beispiel trifft dies zu auf das Problem der PP-Anbindung, das durch automatisches Training auf lokalen, lexikalisierten Features womöglich sogar besser gelöst werden kann als durch explizites Vorhersehen aller plausiblen Kollokationen durch einen Linguisten. Umgekehrt können aber auch offensichtliche Schwächen wie etwa in den n -Gramm-Vorhersagen des POS-Taggers durch manuell formulierte Korrekturregeln wesentlich verbessert werden und die Gesamtleistung wiederum erhöhen. Hybridisierung empirischer und axiomatischer Komponenten erlaubt hier also, den Arbeitsaufwand angemessen zwischen Datensammlung und Regelentwurf zu verteilen. Während solche Spezialprobleme das implementierte Sprachmodell in einzelnen Punkten gezielt verbessern, also tatsächliche Modellfehler von WCDG vermindern, haben andere Komponenten die Funktion von reinen Heuristiken: sie erlauben dem Parser, früher diejenigen Hypothesen zu erreichen, die zwar bereits als optimal definiert sind, deren Optimalität aber praktisch nur mit unvertretbarem Aufwand nachgewiesen werden könnte. Hierdurch können also Suchfehler vermieden werden.

Schließlich können heuristische Prädiktoren auch dazu verwendet werden, den sehr hohen Ressourcenbedarf des WCDG-Parsers zu verringern. Heuristische Optimierung trägt stets den Charakter eines trade-off zwischen Laufzeit und Präzision; 'leistungssteigernde', d.h. beschleunigende Mittel können also entweder eingesetzt werden, um eine gewünschte Parsingqualität schneller zu erreichen, oder aber bei gleichem Ressourcenbedarf bessere Ergebnisse zu erzielen.

Es muß noch einmal betont werden, daß sowohl die implementierte Grammatik des Deutschen als auch die verwendeten Prädiktoren lediglich den Charakter von Existenzbeweisen und nicht etwa optimierten Anwendungssystemen tragen. Bei der WCDG-Entwicklung war das Ziel, nachzuweisen, daß ein handgeschriebenes symbolisches System geeignet ist, eine vollständige menschliche Sprache mit guter Präzision und hoher Abdeckung zu parsen, da dies bei der derzeitigen Vorherrschaft automatisch induzierter Parser zuweilen in Zweifel gezogen wird. Die Hilfskomponenten wiederum wurden bewußt einfach gehalten und bleiben in allen Fällen hinter dem Stand der Kunst zurück. Es ging hier vielmehr darum nachzuweisen, daß die komplementären Eigenschaften solcher Prädiktoren einem 'tiefen' Parser von Nutzen sein können, selbst wenn sie ihre eigene Aufgabe nicht optimal lösen. Daß trotz alledem eine bisher unerreichte Parsinggenauigkeit für das Deutsche demonstriert werden konnte und auf beiden Seiten der Verarbeitung noch erhebliches Verbesserungspotential anzunehmen ist, stützt die These, daß auf absehbare Zeit eine dem Menschen vergleichbare Sprachverarbeitung nur durch die Ausnutzung *aller* bekannten maschinellen Methoden möglich ist.

Literatur

- [BSK97] Thorsten Brants, Wojciech Skut und Brigitte Krenn. Tagging Grammatical Functions. In *Proc. Conference on Empirical Methods in Natural Language Processing (EMNLP-97)*, Providence, RI, USA, 1997.
- [Fot99] Kilian A. Foth. Transformationsbasiertes Constraint-Parsing. Diplomarbeit, Universität Hamburg, Fachbereich Informatik, 1999.
- [GHH⁺82] Barbara Grosz, Norman Haas, Gary Hendrix, Jerry Hobbs, Paul Martin, Robert Moore, Jane Robinson und Stanley Rosenschein. DIALOGIC: a core natural-language processing system. In *Proc, 9th conference on Computational linguistics (COLING-82)*, Seiten 95–100, , Czechoslovakia, 1982. Academia Praha.
- [Niv03] Joakim Nivre. An Efficient Algorithm for Projective Dependency Parsing. In *Proc. 4th International Workshop on Parsing Technologies, IWPT-2003*, Seiten 149–160, 2003.
- [ZZ05] Daniel Zeman und Zdenek Zabokrtsky. Improving Parsing Accuracy by Combining Diverse Dependency Parsers. In *Proc. IWPT*, 2005.

Kilian Foth wurde 1972 in Hamburg geboren und schloß dort das Abitur sowie den Zivildienst ab. Er studierte an der Universität Hamburg Informatik mit dem Schwerpunkt sprachverarbeitender Systeme unter Prof. Dr.-Ing. Wolfgang Menzel, sowie im Nebenfach Anglistik. Nach dem abgeschlossenen Diplom war er als wissenschaftlicher Mitarbeiter in Lehre und Forschung tätig, dort insbesondere in den DFG-Projekten DAWAI und Partial Parsing. Seit 2006 ist er als Lead Developer für Enterprise Thinkware bei der smartSpeed GmbH, Hamburg-Wandsbek beschäftigt. Seit 1987 ist Kilian Foth Organist der Mennonitengemeinde zu Hamburg und Altona.