

Informatik-Fachberichte 313

Herausgeber: W. Brauer
im Auftrag der Gesellschaft für Informatik (GI)

Subreihe Künstliche Intelligenz

Mitherausgeber: C. Freksa
in Zusammenarbeit mit dem Fachbereich 1
„Künstliche Intelligenz“ der GI

Stephan Busemann

Generierung natürlicher Sprache mit Generalisierten Phrasenstruktur- Grammatiken



Springer-Verlag

Berlin Heidelberg New York London Paris

Tokyo Hong Kong Barcelona Budapest

Autor

Stephan Busemann
DFKI GmbH
Stuhlsatzenhausweg 3
W-6600 Saarbrücken 11

CR Subject Classification (1992): I.2.7

ISBN-13: 978-3-540-55818-7

e-ISBN-13: 978-3-642-77714-1

DOI: 10.1007/978-3-642-77714-1

Dieses Werk ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere die der Übersetzung, des Nachdrucks, des Vortrags, der Entnahme von Abbildungen und Tabellen, der Funksendung, der Mikroverfilmung oder der Vervielfältigung auf anderen Wegen und der Speicherung in Datenverarbeitungsanlagen, bleiben, bei auch nur auszugsweiser Verwertung, vorbehalten. Eine Vervielfältigung dieses Werkes oder von Teilen dieses Werkes ist auch im Einzelfall nur in den Grenzen der gesetzlichen Bestimmungen des Urheberrechtsgesetzes der Bundesrepublik Deutschland vom 9. September 1965 in der jeweils geltenden Fassung zulässig. Sie ist grundsätzlich vergütungspflichtig. Zuwiderhandlungen unterliegen den Strafbestimmungen des Urheberrechtsgesetzes.

© Springer-Verlag Berlin Heidelberg 1992

Satz: Reproduktionsfertige Vorlage vom Autor / Herausgeber
33/3140-543210 – Gedruckt auf säurefreiem Papier

Vorwort

Linguistische Grammatiktheorien bilden langfristig eine unverzichtbare Grundlage für die Verarbeitung natürlicher Sprache, will man flexible und vielfältig einsetzbare natürlichsprachliche Systeme entwickeln. Aus informatischer Sicht erscheinen sehr viele linguistische Formalismen zunächst nicht verwendbar, weil sie sich nicht ausreichend effizient implementieren lassen.

Am Beispiel der linguistischen Syntaxtheorie der Generalisierten Phrasenstruktur-Grammatik (GPSG) und des Problems der Generierung geschriebener natürlicher Sprache zeigt dieses Buch, wie die Brücke zwischen informatischen und linguistischen Forderungen geschlagen werden kann, so daß eine nutzbringende Symbiose entsteht.

Das Thema des Buches ist interdisziplinär: Fragen der linguistischen Theoriebildung werden in Bezug gesetzt zu Problemen der effizienten Sprachverarbeitung. Das Buch führt in die Theorie der GPSG ausführlich ein, diskutiert sie sowohl aus linguistischer wie aus informatischer Sicht und münzt die Kritik um in einen effizient implementierten Formalismus. Auf dieser linguistischen Grundlage werden Algorithmen zur Sprachgenerierung vorgestellt und bewertet. Die Abbildung von semantischen Ausgangsstrukturen auf syntaktische GPSG-Strukturen erfolgt mit Hilfe von getrennt repräsentierten Produktionsregeln, wie sie aus den Produktionensystemen der Künstlichen Intelligenz bekannt sind.

Das Buch wendet sich an Informatikerinnen¹, die an Problemen der Verarbeitung natürlicher Sprache interessiert sind, sowie an Computerlinguistinnen und Linguistinnen mit einem guten Hintergrund in Informatik. Für manche Leserinnen wird das Glossar wichtiger linguistischer Fachbegriffe nützlich sein (die darin enthaltenen Begriffe sind im Text bei ihrem ersten Vorkommen durch einen Pfeil (→) gekennzeichnet).-

Dieses Buch ist eine überarbeitete Fassung meiner Dissertation, die im Juli 1990 von der Universität des Saarlandes angenommen wurde. Die Arbeit entstand zum überwiegenden Teil innerhalb der Projektgruppe KIT (Künstliche Intelligenz und Textverstehen) am Fachbereich Informatik der TU Berlin zwischen 1985 und 1990.² Sie wurde am Deutschen Forschungszentrum für Künstliche Intelligenz in Saarbrücken fertiggestellt.

Meine Beschäftigung mit GPSG erfolgte im Rahmen der Projekte KIT-NASEV und KIT-FAST. Das Projekt KIT-NASEV entwickelte neue Analyse- und Syntheseverfahren für die maschinelle Übersetzung, wobei eine zentrale Fragestellung war, wie GPSG überhaupt für Parsing und Generierung nutzbar gemacht werden kann. Im daran anschließenden Projekt KIT-FAST wurden Funktor-Argument-Strukturen für den Transfer hinzugefügt. Damit konnte maschinelle Übersetzung auf der Grundlage eines Transferansatzes mit Hilfe einer oberflächennahen, satzsemantischen Repräsentationssprache realisiert werden.

In diesem Buch greife ich wesentliche Ergebnisse der Forschung in KIT-NASEV und KIT-FAST auf. Viele Kolleginnen haben zum Gesamtergebnis beigetragen und hier beschriebene Teile mitgestaltet. Im Rahmen des Projektes KIT-NASEV wurde eine Version des GPSG-Formalismus entworfen

¹Gemeint sind selbstverständlich die „Informatikerinnen und Informatiker“. Um solche klobigen Formulierungen (oder orthographische Purzelbäume wie in „InformatikerInnen“) zu vermeiden, verwende ich durchgehend die feminine Form, wenn an beiderlei Geschlecht gedacht ist.

²KIT umfaßt Projekte aus verschiedenen Teildisziplinen der KI und der Computerlinguistik, was einen regelmäßigen, fruchtbaren Gedankenaustausch zwischen den Kolleginnen ermöglicht.

und implementiert, die für Sprachverarbeitung besonders gut geeignet ist. Daran haben maßgeblich Christa Hauenschild, James Kilbury, Wilhelm Weisweber, William Keller und ich mitgewirkt. Auf der Grundlage dieses Formalismus habe ich zwei Verfahren entworfen und implementiert, die zeigen, wie mit GPSG geschriebene natürliche Sprache generiert werden kann. Die benutzten Grammatik-Fragmente des Deutschen und Englischen wurden hauptsächlich von Susanne Preuß und Margaret Garry im Rahmen der genannten Projekte geschrieben und von Susanne Preuß und mir getestet. Die verwendeten Funktor-Argument-Struktur-Fragmente entwickelten federführend Christa Hauenschild und Carla Umbach.-

Ich danke meinem Doktorvater Wolfgang Wahlster für seine Unterstützung über einen langen Zeitraum hinweg und seine Aufmunterung, die besonders in den schwierigen Phasen der Arbeit wichtig war. Hans Uszkoreit, dem zweiten Gutachter meiner Dissertation, danke ich für zahlreiche Hinweise, Ideen und spannende Diskussionen im Bereich der Unifikationsgrammatiken, die mir viele neue Perspektiven erschlossen.

Von prägendem Einfluß war für mich die enge Zusammenarbeit mit Christa Hauenschild in KIT-FAST. Christa verdanke ich wichtige Zugänge zur formalen Linguistik; mit ihr über GPSG, Generierung und die wichtigen Dinge des Lebens zu diskutieren, war nicht nur außerordentlich anregend, sondern es machte auch ganz einfach Freude. Susanne Preuß, Carla Umbach und Wilhelm Weisweber bin ich für viele Anregungen und kritische Fragen dankbar, die mich immer wieder zum Überdenken meiner Grundannahmen brachten.

Allen Mitgliedern der Projektgruppe KIT an der TU Berlin und dem Leiter der KIT-Projekte Bernd Mahr danke ich für ihre freundschaftliche Unterstützung und Geduld, ohne die die Arbeit in dieser Form wohl nicht zustande gekommen wäre. Besonders wichtig waren Kai von Lucks unermüdliches Drängen und seine Antriebskraft, die bis zu seinem Weggang aus Berlin diese Arbeit mit in Gang brachten. Bei der Implementation des Berliner GPSG-Systems bildete Michael Eimerachers Tracepaket für Waterloo-Core-Prolog eine wichtige Hilfestellung. Christof Peltason erleichterte mit zahlreichen Service-Programmen die Benutzung eines KIT-weiten Netzes verschiedenster Rechner, wovon ich sehr profitiert habe. Cordula Lippke leistete wertvolle technische Hilfe bei der Erstellung von Graphiken.

Dem Deutschen Forschungszentrum für Künstliche Intelligenz in Saarbrücken und meinen Kolleginnen an meinem dortigen Arbeitsplatz bin ich dankbar für die Möglichkeit, die Arbeit abzuschließen.

Mein besonderer Dank gilt Katharina Morik für ihre freundschaftlichen Anstöße und Ideen sowohl in fachlicher als auch in ganz menschlicher Hinsicht. Schließlich danke ich meiner Frau Monika, die so oft auf gemeinsam verbrachte Zeit verzichtete und mir doch immer wieder Kraft zum Weitermachen gab.

S.B.

Saarbrücken,
Juni 1992

Inhaltsverzeichnis

1	Einführung	1
1.1	Das Thema	1
1.2	Fragestellungen und Ergebnisse	3
1.3	Übersicht	5
2	Maschinelle Generierung natürlicher Sprache	7
2.1	Generierungsansätze in KI und Psycholinguistik	7
2.2	Computerlinguistische Ansätze	9
2.2.1	Das Generierungsproblem in computerlinguistischen Ansätzen	10
2.2.2	Probleme mit Top-Down-Verfahren am Beispiel von HPSG	11
2.2.3	Inkrementelle Generierung mit TAGs und mit Segmentgrammatiken	12
2.2.4	Ein strukturgetriebenes Generierungsverfahren am Beispiel von LFG	14
2.2.5	Bidirektionale Bottom-Up-Verarbeitung mit Earley-basierter Deduktion	15
2.2.6	Steuerung durch den semantischen Head am Beispiel von DCG und UCG	17
2.3	Einordnung der vorliegenden Arbeit	18
3	Generalisierte Phrasenstrukturgrammatiken	21
3.1	Zur Entwicklung von GPSG	22
3.1.1	Zur Beschreibung natürlicher Sprachen mit kontextfreien Grammatiken	23
3.1.2	Der indirekte, metagrammatische Ansatz	24
3.1.3	Der direkte, constraint-basierte Ansatz	29
3.1.4	GPSG und das Lexikon	33
3.2	Eine formale Definition von GPSG	34
3.3	Die axiomatische Version von GPSG	38
3.3.1	Merkmalspezifikationen und Kategorien	38
3.3.2	Feature Cooccurrence Restrictions und Feature Specification Defaults	39
3.3.3	Das ID/LP-Format	40
3.3.4	Metaregeln	44
3.3.5	Zulässige Bäume	46
3.3.6	Das Foot-Feature-Prinzip	47
3.3.7	Das Control-Agreement-Prinzip	50
3.3.8	Die Head-Feature-Konvention	57
4	Eine konstruktive Version von GPSG	63
4.1	Probleme der Algorithmisierung von GPSG	64
4.2	Der Berliner GPSG-Formalismus	68
4.2.1	Kategorien, Extension und Unifikation	68
4.2.2	FCRs, ID-Regeln und LP-Aussagen	69
4.2.3	Die Head-Feature-Konvention	70
4.2.4	Das Agreement-Prinzip	71
4.2.5	Das Foot-Feature-Prinzip	73

4.2.6	Eine Anwendungsreihenfolge	73
4.2.7	Zulässige Bäume	74
4.2.8	Die wesentlichen Unterschiede zwischen dem axiomatischen und dem konstruktiven GPSG-Formalismus	76
4.3	Die Architektur des Berliner GPSG-Systems	77
4.4	Die Grammatikfragmente für Deutsch und Englisch	80
4.5	Prozedurale Aspekte der Strukturbildung	84
4.5.1	Verarbeitungsstrategien aufgrund der konstruktiven GPSG	85
4.5.2	Instantiierung durch Konstruktion und HFC	87
4.6	Der Lexikonzugriff bei der Generierung	89
4.6.1	Generierung mit Stammformenlexika	89
4.6.2	Paradigmatische Lücken	90
4.6.3	Die Generierung von Perfekt-Hilfsverben im Deutschen	91
4.7	Weitere Implementationen von GPSG	92
4.7.1	Das ProGram-System von Evans	94
4.7.2	Der GPSG-Parser von Naumann	94
4.7.3	Bitvektor-Repräsentationen: Nakazawa und Neher	95
4.7.4	Propagierungsregeln: Phillips und Thompson	97
4.8	Indirekte und direkte Interpretationen von GPSG	99
4.8.1	Die stark direkte Interpretation von GPSG	100
4.8.2	Indirekte und schwach direkte Interpretationen von GPSG	101
5	Die Ausgangsstrukturen der Generierung	103
5.1	Syntaktische Ausgangs-Strukturen	103
5.2	Funktor-Argument-Strukturen	106
5.2.1	Der Formalismus	107
5.2.2	Die in FAS repräsentierte Information	108
6	Generierung mit GPSG	113
6.1	Verarbeitungsstrategien aufgrund der Ausgangsstrukturen	113
6.2	Grammatikgetriebene Generierung	115
6.2.1	Das Verfahren	115
6.2.2	Ein Beispiel	117
6.2.3	Diskussion	126
6.3	Strukturgetriebene Generierung	128
6.3.1	Das Verfahren	129
6.3.2	Produktionensysteme	132
6.3.3	Pattern-Action-Regeln	134
6.3.4	Die Abbildungen von FAS-Ausdrücken auf GPSG-Strukturen	137
6.3.5	Die Anwendungsfolge von Pattern-Action-Regeln	142
6.3.6	Die Generierung von dislozierten Konstituenten	145
6.3.7	Die Erzeugung des Oberflächenkasus	147
6.3.8	Ein Beispiel	149
6.3.9	Diskussion	159
7	Zusammenfassung und Ausblick	163
7.1	Zusammenfassung	163
7.2	Die Trennung von Verarbeitungsstrategien, Formalismus und Grammatiken	164
7.3	Ausblick	164

A	GPSG-Syntax für das deutsche Fragment	169
A.1	Merkmale	169
A.2	ID-Regeln	169
A.3	LP-Aussagen	171
A.4	Feature-Cooccurrence-Restrictions	171
A.5	Einige Aliasbezeichner	171
B	Pattern-Action-Regeln für das deutsche Fragment	173
B.1	FAS-Kategorien	173
B.2	Pattern-Action-Regeln	174
	Bibliographie	179
	Glossar linguistischer Fachbegriffe	191
	Index	199

Abbildungsverzeichnis

1.1	Eine syntaktische GPSG-Struktur	3
2.1	Die Modularisierung des Generierungsprozesses	9
2.2	Subkategorisierung in HPSG	11
2.3	Initialer Baum (a), auxiliärer Baum (b) und Adjunktion beider (c) in TAG	12
2.4	Konkatenation (a) und Gabelung (b) in Segmentgrammatiken	13
2.5	c-Struktur (a), entfaltete f-Struktur (b) und Ausgangs-f-Struktur (c) in LFG	15
3.1	Merkmalspezifikationen zur Beschreibung von N, V, A und P	25
3.2	Die Organisation der mittleren GPSG	29
3.3	Die Organisation der späten GPSG	32
3.4	Topologische Bereiche bei Füller-Lücke-Konstruktionen	48
3.5	Unerlaubtes Zusammentreffen von slash-Spezifikationen	49
3.6	Kontrollbeziehungen zwischen Kategorien in lokalen Bäumen	51
3.7	CAP und das Zusammenwirken der Merkmalinstantiierungsprinzipien	55
3.8	Projektionen aufgrund der Topikalisierungsregel	60
3.9	HFC und das Zusammenwirken der Merkmalinstantiierungsprinzipien	61
4.1	Kongruenzbeziehungen mit dem revidierten AP	72
4.2	Anwendungsreihenfolge in einer konstruktiven Version von GPSG	74
4.3	Kontrolle der Baumgenerierung durch Constraints	75
4.4	Der Aufbau des Berliner GPSG-Systems	78
4.5	Der Hilfsverbkomplex in der deutschen Grammatik	83
4.6	Der NP-Komplex in der deutschen Grammatik	84
4.7	Strukturbildung und Merkmalsinstantiierung	86
4.8	Strukturbildung bei der Generierung	88
4.9	Zur lexikalischen Beschränkung syntaktischer Strukturen	91
4.10	Bitvektor für die in allen Merkmalen undefinierte Kategorie	96
4.11	HFC als Operation über Bitvektoren	97
4.12	GPSG-Systeme im Überblick	102
5.1	BNF-Definition für SAS (deutsch)	104
5.2	Ein SAS-Ausdruck für den Satz (5.1)	105
5.3	FAS-Ausdruck für einen deutschen Satz	108
6.1	Generierung einer GPSG-Struktur aus einem SAS-Ausdruck	118
6.2	Architektur des strukturgetriebenen Systems	130
6.3	Eine Pattern-Action-Regel zur Abbildung eines Terms auf eine NP	135
6.4	Anwendung von Pattern-Action-Regeln auf einen lokalen FAS-Baum.	138
6.5	Abbildung einer hierarchischen in eine flache Struktur.	141
6.6	Die Generierung von Oberflächenkasus	148
6.7	Lexikalische Pattern-Action-Regel für Verben	149