

Interpretation und Artikulation mit Äußerungs-Bedeutungs-Transduktoren

Von der Fakultät für MINT - Mathematik, Informatik, Physik,
Elektro- und Informationstechnik
der Brandenburgischen Technischen Universität Cottbus –Senftenberg
zur Erlangung des akademischen Grades eines
Doktors der Ingenieurwissenschaften
(Dr.-Ing.)

genehmigte Dissertation
vorgelegt von

M. Sc.

Jens Lindemann

geboren am 16.01.1979 in Berlin

Gutachter: Prof. Dr.-Ing. habil. Matthias Wolff
Gutachter: Prof. Dr. Günther Wirsching
Gutachter: Prof. Dr.-Ing. Klaus Fellbaum

Tag der mündlichen Prüfung: 02. Juni 2016

Inhaltsverzeichnis

Abstract	xi
Kurzfassung	xii
1 Einleitung	1
2 Sprachkommunikation	7
2.1 Sprache als Kommunikationsform	7
2.2 Gebärdensprache	8
2.2.1 Phonologie der Gebärdensprache	8
2.2.2 Grammatische Besonderheiten der Deutschen Gebärdensprache	10
2.2.3 Gebärdensprach-Notationssysteme	11
2.3 Zur Bedeutungsrepräsentation in sprachverarbeitenden Systemen	12
2.4 Sprachmodelle	14
2.4.1 Formale Grammatiken	14
2.4.2 Endliche Automaten	16
2.4.3 Statistische Sprachmodelle	22
3 Verfahren zur Interpretation von gesprochener Sprache und zur Generierung von Wortfolgen	23
3.1 Semantische Interpretation von gesprochener Sprache	23

3.1.1	Syntaxbasierte semantische Interpretation	25
3.1.2	Wissensbasierte Interpretation mit semantischen Grammatiken .	27
3.1.3	Statistikbasierte Interpretationsverfahren	28
3.2	Artikulation von Sprachzeichenfolgen	45
3.2.1	Artikulation mittels vorgefertigter Textschablonen	45
3.2.2	Standardmodell der regelbasierten Sprachgenerierung	46
3.2.3	Verwendung statistischer Verfahren zur Artikulation	47
4	Äußerungs-Bedeutungs-Transduktoren	51
4.1	Merkmal-Werte-Relationen als Bedeutungsrepräsentation	52
4.2	Semantische Grammatiken	54
4.2.1	Konzept des bedeutungsorientierten Sprachmodells	54
4.2.2	Hierarchie von Grammatiken	56
4.2.3	Bezug zur Interpretation in einem Sprachdialogsystem	56
4.3	Interpretation mit endlichen Äußerungs-Bedeutungs-Transduktoren . .	60
4.4	Artikulation mit endlichen Äußerungs-Bedeutungs-Transduktoren . . .	65
4.5	Zum Einsatz von Petrinetztransduktoren in der semantischen Sprachverarbeitung	69
5	Modellierung von Grammatiken für ein semantisches Schema	73
5.1	Beschreibung der Bedeutungseinheiten zur Elementargrammatik	73
5.2	Sprachmaterial zur semantischen Modellierung	76
5.3	Modellierung von Äußerungs-Bedeutungs-Transduktoren	81
5.3.1	Erstellung eines Äußerungs-Bedeutungs-Transduktors zur Elementargrammatik	81
5.3.2	Platzhalter in mikrolokalen Grammatiken	84

5.3.3	Modellierung eines Äußerungs-Bedeutungs-Transduktors zur mikrolokalen Grammatik	85
5.3.4	Bewertung und Diskussion des Modellierungsergebnisses	91
6	Ergebnisse der semantischen Interpretation und Artikulation für das modellierte Schema	94
6.1	Semantische Interpretation	94
6.1.1	Interpretation mit Äußerungs-Bedeutungs-Transduktor	95
6.1.2	Interpretation mit konzeptuellem Sprachmodell	96
6.1.3	Vergleich der Interpretationsvorgänge	99
6.2	Artikulation einer Bedeutungsstruktur mit dem inversen Äußerungs-Bedeutungs-Transduktor	101
6.3	Wortfolgenbewertung durch ein Sprachmodell	103
6.3.1	Verwendung eines mikrolokalen Sprachmodells	104
6.3.2	Normierte Gesamtbewertung	106
6.3.3	Microsoft Web N-gram Sprachmodell	107
6.3.4	Ergebnisse der normierten Bewertung von generierten Wortfolgen	108
7	Zusammenfassung und Ausblick	109
7.1	Zusammenfassung	109
7.2	Weitere Untersuchungs- und Anwendungsmöglichkeiten	111
Anhang		
A	Verwendetes Sprachmaterial	115
A.1	Lautsprachäußerungen zur Modellierung der Elementargrammatik . . .	115
A.2	Zusätzliche lautsprachliche Äußerungen	116
A.3	Verwendete Glossennotationen zur Modellierung der Gebärdensprachgrammatiken	119

A.4	Gebärdensprachtranskriptionen	120
B	Automatengraphen erstellter Transduktoren	123
C	Generierte Sprachzeichenfolgen	128
C.1	Artikulierte Glossensequenzen	128
C.2	Ergebnisse der Sprachmodellbewertung	129
C.2.1	Rangfolge durch normierte Bewertung (Bigramm-Sprachmodell)	129
C.2.2	Rangfolge der Gesamtbewertung durch das Microsoft Web N-gram Sprachmodell	132
C.2.3	Rangfolge durch Normierung der Bewertung des Microsoft Web N-gram Sprachmodells	134
D	Nicht öffentlich zugängliche Quellen	137
D.1	Korrespondenz mit Prof. Horst Ebbinghaus [Ebb15]	137
D.2	Korrespondenz mit Prof. Rachel Rosenstock [Ros15]	139
	Literaturverzeichnis	140

Abbildungsverzeichnis

1.1	Vereinfachte Systemstruktur eines intelligenten hierarchischen Signalverarbeitungssystems nach [HW15]	2
1.2	Funktionalität der Beispielanwendung - bidirektionale Übersetzung zwischen beschrifteten Totalordnungen und beschrifteten partiellen Ordnungen	4
2.1	Manuelle und nichtmanuelle Gebärdensprachparameter	9
2.2	Zustandsübergang eines gewichteten Transduktors	17
2.3	Visualisierung eines ungewichteten Automaten als Graphen	18
3.1	Grundprinzip der syntaxgesteuerten semantischen Analyse nach [JM00]	26
3.2	Syntaxbaum mit semantischer Etikettierung nach [MBH ⁺ 08]	26
3.3	Beispiel eines semantischen Ableitungsgraphen nach [BSH08]	27
3.4	Zweistufiger Interpretationsprozess - Attribut-Wert-Extraktion zu der klassifizierten Konzeptmarkierungsfolge $\hat{\mathbf{t}}$	29
3.5	Konzeptmarkierungssequenz zu der Wortfolge „ <i>Peter trinkt das Bier</i> “ .	30
3.6	Vereinfachtes graphisches Modell eines Linear-Chain Conditional Random Fields	31
3.7	Graphische Struktur eines Maximum-Entropie Markovmodells	32
3.8	Ableitung eines semantischen Baums zu positiv klassifizierter Tupelmengen	34
3.9	Zeitlich „ausgerolltes“ rekurrentes Neuronales Netz (Elman-Architektur)	34
3.10	Prinzip eines log-linearen Übersetzungsmodells nach [Mac09]	36

3.11	Vergleich eines möglichen semantischen Ableitungsgraphen und der dazugehörigen Sequenz von semantischen Vektoren	37
3.12	Grundprinzip der Konstruktion des Automatengraphen \mathcal{A}_{WK} zum Worthythesengraph \mathcal{A}_W nach [Cal15]	43
3.13	Prinzipielle Struktur des Konsensmodells	46
4.1	System zur gemeinsamen Interpretation und Artikulation mittels Äußerungs-Bedeutungs-Transduktoren (Übersicht) nach [HW15]	51
4.2	Ungewichtete Merkmal-Werte-Relation	53
4.3	Sequenzieller Ablauf eines Dialogschritts - visualisiert am Blockschaltbild des Dialogsystems	57
4.4	Zusammenhang zwischen referenziellen semantischen Anker und zugeordneten semantischen Anker	60
4.5	Äußerungs-Bedeutungs-Transduktoren zur Interpretation von sprachlichen Realisierungen	61
4.6	MWR als Baumstruktur zum Beispiel Bahnauskunftssystem [Lin14]	64
4.7	Visualisierung der Konstruktion möglicher Permutationen	66
4.8	Hierarchieebene innerhalb einer MWR	67
4.9	Äußerungs-Bedeutungs-Transduktoren zur Artikulation von Sprachzeichenfolgen	68
4.10	Graph eines einfachen PNT mit der Anfangsmarkierung $p_I = m_0$	71
5.1	Visualisierung der Beispieläußerung	74
5.2	Beispiel einer Gebärdensprachtranskription	79
5.3	Gebärdensprachtranskription der Beispieläußerung 6	80
5.4	Graph des Transduktors zur DGS-Beispieläußerung 5	81
5.5	Graph des „Root-FST“ zur Erstellung eines endlichen Äußerungs-Bedeutungs-Transduktors zum semantischen Schema „PERS-ACT-LOC“	82
5.6	Automatengraph des endlichen Äußerungs-Bedeutungs-Transduktors zur gepärdensprachlichen Elementargrammatik	83
5.7	Verbindliche Merkmale des modellierten semantischen Schemas	85

5.8	FST für Platzhalter „loc_rel“ (DGS)	86
5.9	FST für Platzhalter „action“ (DGS)	87
5.10	FST für Vornamen (Lautsprache)	88
5.11	FST für Familiennamen (Lautsprache)	88
5.12	Initialer FST für den Platzhalter „name“ (Lautsprache)	90
6.1	Transduktor T_W für die ausgewählte Wortfolge	95
6.2	Interpretationsergebnis - Ausschnitt des Transduktorgraphen T_{MWR} . .	95
6.3	Merkmal-Werte-Relation als Ergebnis des Interpretationsprozesses . . .	96
6.4	Darstellung der Übersetzungsprozesse einzelner Transduktoren am Beispiel des Sprachzeichens „Peter“.	98
6.5	Ausschnitt des Transduktors $T_W \circ T_G \circ T_{G2K} \circ T_{KSM}$ zur Beispieläußerung „ <i>Wartend steht Peter dort drüben.</i> “	98
6.6	Interpretationsergebnis als gewichtete Merkmal-Werte-Relation	101
6.7	Zusammenhang zwischen semantischem Anker in einer Merkmal-Werte-Relation und der dazugehörigen Symbolfolge aus der MWR-Zeichenkette	102
B.1	FST für Platzhalter „loc_rel“ (Lautsprache)	123
B.2	FST für Platzhalter „name“ (DGS)	123
B.3	FST für Platzhalter „action“ (Lautsprache)	124
B.4	Unterteilung des Platzhalters „name“ (Lautsprache) unter Berücksichtigung des Geschlechts einer Person	125
B.5	Transduktor T_{MWP} zur interpretierten Sprachäußerung „ <i>Wartend steht Peter dort drüben.</i> “	126
B.6	Ausschnitt des Transduktors T_{G2K}	127

Abstract

The use of finite state transducers for bidirectional translation between the syntactic and semantic level in a speech dialogue system will be investigated in the present PhD thesis. For meaning representation so-called *feature-values relations* are used. This data structure can be illustrated by an acyclic directed graph. Such labeled partial orders enable a hierarchical conceptual meaning representation.

An approach for modeling *utterance meaning transducers* is presented, which is based on a meaning-oriented language model. An essential prerequisite for this is a corpus of utterances, which can be collected, for instance, by *Wizard of Oz* experiments.

For own investigations *finite state transducers* are developed for utterances of spoken German language and German Sign Language. These semantic grammars are used for both spoken language understanding and language generation. Thus, at a given word lattice possible semantic strings can be generated automatically and this bracketed semantic string can be converted to the appropriate *feature-values relation*. In an application example a sequence of words is directly generated from this meaning representation. Inverse *utterance meaning transducers* are used for this task, but first the *feature-values relation* has to be linearized. As a result of these two translation processes, a paraphrase to a given sequence of words will be obtained. It is also feasible to realize a meaning-oriented translation of a given utterance by using different language-dependent semantic grammars for spoken language understanding and generation.

In addition, a current method of semantic decoding has been implemented, which is also based on finite state transducers. This approach uses a conceptual stochastic language model. Therefore, it is possible to compare the two different methods.

Kurzfassung

In der vorliegenden Arbeit wird die Verwendung von endlichen Transduktoren zur bidirektionalen Übersetzung zwischen der syntaktischen und semantischen Ebene eines natürlichsprachlichen Dialogsystems untersucht. Zur Bedeutungsrepräsentation werden *Merkmals-Werte-Relationen* verwendet. Solche beschrifteten partiellen Ordnungen ermöglichen eine hierarchische Strukturierung von semantischen Begriffen und können durch gerichtete azyklische Graphen dargestellt werden.

Auf Grundlage eines bedeutungsorientierten Sprachmodells wird eine mögliche Vorgehensweise zur Modellierung von *Äußerungs-Bedeutungs-Transduktoren* vorgestellt. Voraussetzung dafür ist eine Sprachdatensammlung, welche zum Beispiel mit Hilfe von *Wizard-of-Oz* Experimenten erstellt werden kann.

Für die eigenen Untersuchungen wurden Transduktoren zu Sprachäußerungen der deutschen Lautsprache und der Deutschen Gebärdensprache konstruiert. Diese semantischen Grammatiken werden sowohl für die Interpretation als auch für die Artikulation eingesetzt. Auf der Perzeptionsseite ist somit die automatische Generierung von semantischen Zeichenketten zu einem gegebenen Worthypothesegraphen realisierbar. Durch die Klammerstruktur der semantischen Zeichenketten kann die dazugehörige *Merkmals-Werte-Relation* erzeugt werden. Diese Bedeutungsrepräsentation wird in der Beispielanwendung unter Verwendung des inversen *Äußerungs-Bedeutungs-Transduktors* direkt wieder artikuliert. Zuvor ist jedoch eine Linearisierung der *Merkmals-Werte-Relation* notwendig. Als Ergebnis dieser beiden Übersetzungsprozesse erhält man eine sinngemäße Wiedergabe zu der gegebenen Sprachzeichenfolge. Durch die Verwendung sprachabhängiger semantischer Grammatiken ist zudem auch eine bedeutungsorientierte Übersetzung des erkannten sprachlichen Ausdrucks realisierbar.

Zusätzlich wurde auch ein aktuelles Verfahren zur semantischen Dekodierung mit endlichen Transduktoren umgesetzt, welches ein konzeptuelles stochastisches Sprachmodell nutzt. Damit ist eine vergleichende Bewertung dieser beiden Vorgehensweisen zur semantischen Interpretation möglich.

Kapitel 1

Einleitung

Sprache ist die wichtigste und natürlichste Kommunikationsform. Sie dient zum Informationsaustausch und ist Träger von semantischen Informationen. Sinnvoll ist deshalb auch die Möglichkeit der Verwendung von Sprache bei der Mensch-Maschine-Interaktion. Auf diese Weise kann beispielsweise eine einfache und intuitive Steuerung komplexer elektronischer Geräte im Haushalt¹ erreicht werden. Einen breiten Anwendungsbereich gibt es zudem auch für sprachbasierte Informations- und Auskunftssysteme. Solche technischen Systeme sollten sinnvolle Sprachäußerungen des Benutzers „verstehen“ können und adäquat darauf reagieren. Ist eine Mensch-Maschine-Schnittstelle multimodal gestaltet, so kann potenziell neben einer intuitiven Bedienbarkeit auch eine Berücksichtigung der Bedürfnisse von älteren Menschen sowie Menschen mit Behinderungen erreicht werden.

Einordnung der Arbeit

Die vorliegenden Untersuchungen erfolgen im Rahmen des Projektes *UASR* (Unified Approach to Speech Synthesis and Recognition) [UAS]. Ziel dieses Forschungsvorhabens ist die Entwicklung eines kognitiven hierarchischen Systemmodells, welches nicht nur zur Verarbeitung von Sprachsignalen, sondern auch für beliebige Sensorsignale eingesetzt werden kann [WRH12].

Kognitive dynamische Systeme können durch einen Perzeptions-Aktions-Zyklus beschrieben werden und weisen ein zielgerichtetes Verhalten auf. Neben der Fähigkeit zum Schließen unter Unsicherheit sind auch Lern- sowie Adaptionsvorgänge notwendig [You10]. Durch die automatische Modifikation vorhandener Wissensmodelle und Dialogstrategien ist auch eine Anpassung an das Kommunikationsobjekt möglich. Solche intelligenten Systeme besitzen eine Sensorik, die eine Beobachtung des Zustands der Umwelt ermöglicht. Im Kontext von Kommunikationsprozessen wird nachfolgend

¹ Untersuchungen hierzu werden beispielsweise aktuell im BMBF-Projekt *Universal Cognitive User Interface* (2015 – 2018) [UCU] durchgeführt.

Ein wesentliches Merkmal des UASR-Systems ist die konsequente Nutzung von endlichen Transduktoren (*finite state transducers*). Gewichtete endliche Transduktoren sind vielseitig einsetzbar. Anwendungsgebiete im Bereich der Sprachverarbeitung sind z. B. die Spracherkennung und -synthese sowie die semantische Interpretation und die Informationsextraktion. Auch zur computerunterstützte Übersetzung [BBC⁺09] können endliche Transduktoren verwendet werden.

In dieser Arbeit wird speziell die Interpretation und die Artikulation von Sprachzeichenfolgen in Sprachdialogsystemen untersucht. Dabei wird Sprache im Zusammenhang mit Kommunikationsprozessen betrachtet. Eine Determinierung der Bedeutung von Sprachäußerungen muss deshalb verstärkt auf der kommunikativen Ebene gesucht werden. Somit rückt der Sprachgebrauch in den Fokus. Das verbale Verhalten des Benutzers eines Dialogsystems dient einem bestimmten außersprachlichen Zweck. Auf dieser Ebene sind auch die Intentionen von nichtsprachlichen Verhalten, wie z. B. Mimik, Gestik oder taktile Eingaben zu beschreiben. Zur Bestimmung der Bedeutungsträger zu einer Sprachäußerung (= kodierte Absicht des Sprechers) müssen externe Faktoren, wie beispielsweise die aktuell vorherrschende Interaktionssituation, berücksichtigt werden.

In der Regel werden die zur semantischen Interpretation verwendeten Modelle zwar domänenspezifisch erstellt, jedoch ansonsten kontextunabhängig eingesetzt. Eine mögliche Berücksichtigung der Sprechsituation (z. B. durch Dialogakttyp oder der Dialoghistorie) erfolgt erst durch die Verhaltenssteuerung bzw. das Dialogmanagement.

Dabei ist eine Prädiktion der erwarteten Absicht des Kommunikationsteilnehmers entgegen der Perzeptionsrichtung durchaus sinnvoll [WRH12]. Eine Berücksichtigung der Erwartung bei der Interpretation kann mit dem Einsatz von *Äußerungs-Bedeutungs-Transduktoren* erreicht werden. Gleichzeitig erfolgt dadurch eine stärkere Verflechtung von syntaktischem, semantischem und pragmatischem Wissen.

Als einheitliche Datenstruktur für die semantische und pragmatische Ebene werden in unserem Systemmodell *Merkmal-Werte-Relationen* [HKL⁺09, WHK⁺12] verwendet. Diese beschrifteten partiellen Ordnungen ermöglichen eine hierarchische Strukturierung einer Menge von semantischen Kategorien und dazugehörigen Datenwerten. Verwendete Beschriftungen werden beispielsweise aus vorhandenen Datenbankstrukturen abgeleitet. *Merkmal-Werte-Relationen* ermöglichen somit eine adäquate Repräsentation von Bedeutungen. Durch den Einsatz geeigneter semantischer Grammatiken wird eine direkte Zuordnung solcher Bedeutungsträger zu der analysierten Sprachhandlung erreicht.

Zielstellung

Eine Untersuchung der prinzipiellen Verwendungsmöglichkeit von endlichen Transduktoren für die funktionale Erweiterung des UASR-Systems stellt ein wichtiges Ziel dieser Arbeit dar.

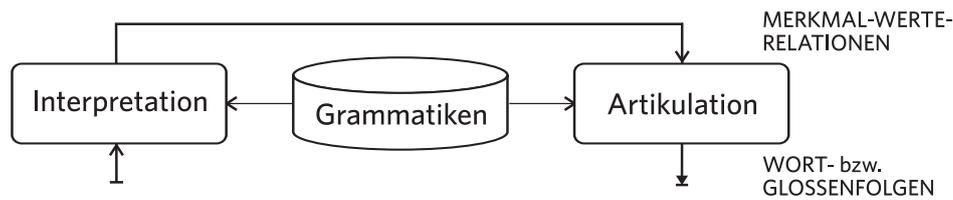


Abb. 1.2: Funktionalität der Beispielanwendung - bidirektionale Übersetzung zwischen beschrifteten Totalordnungen und beschrifteten partiellen Ordnungen

Zu diesem Zweck ist eine geeignete Vorgehensweise zur Erzeugung endlicher Automaten zu entwickeln, welche bidirektionale Übersetzungen zwischen Sprachzeichenfolgen und hierarchischen Bedeutungsrepräsentationen (*Merkmal-Werte-Relationen*) ermöglichen. Die Beispielgrammatiken sollen sowohl für Wortfolgen der deutschen Lautsprache als auch für *Glossennotationen* von Sprachäußerungen der Deutschen Gebärdensprache erstellt werden. *Glossen* sind in Großbuchstaben geschriebene Wörter, welche auf die Bedeutung der Gebärden verweisen. Der Glossenname kann dabei durch zusätzliche Symbole ergänzt werden.

Die auf diese Weise modellierten *Äußerungs-Bedeutungs-Transduktoren* werden in einer Beispielanwendung eingesetzt. In dieser Applikation wird die vorgegebene Sprachzeichenfolge automatisch in eine *Merkmal-Werte-Relation* überführt. Im Anschluss erfolgt die unmittelbare, sprachenabhängige Generierung einer Wortfolge, aus der zuvor erzeugten Semantikrepräsentation. Dieser sequentielle Ablauf der beiden Übersetzungsprozesse ist in Abb. 1.2 dargestellt. Somit wird durch den Gesamtvorgang entweder eine Paraphrasierung der Sprachäußerung oder die bedeutungsgetriebene Übersetzung einer vorgegebenen Zeichenfolge erreicht.

An dieser Stelle sei darauf hingewiesen, dass lediglich Semantikrepräsentationen anhand von Sprachzeichenfolgen abgeleitet werden. Para- und nichtlinguistische Informationen, die in einer Sprechsituation beobachtet werden können, sind somit nicht Gegenstand der Untersuchungen dieser Arbeit.

Ansatz

Das in [WL13] beschriebene Konzept für ein bedeutungsorientiertes Sprachmodell stellt die Grundlage für die hier zu modellierenden semantischen Grammatiken dar. Der Zusammenhang zwischen einer Sprachäußerung und der Absicht des Sprachzeichenbenutzers wird durch kontextabhängige Äußerungs-Bedeutungs-Paare modelliert.

Der Artikulationsvorgang kann prinzipiell als inverser Interpretationsprozess verstanden werden [You02]. Der hier verwendete Ansatz ermöglicht eine bidirektionale Übersetzung zwischen Syntax und Semantik. Aus diesem Grund stellen die erzeugten Grammatiken auch gemeinsame Modelle für Interpretation und Artikulation dar.

Für die Generierung von verbalem Verhalten werden dementsprechend inverse *Äußerungs-Bedeutungs-Transduktoren* eingesetzt. Oft wird mit dem Begriff „Sprache“ direkt die Lautsprache assoziiert. Gebärdensprachen verwenden zwar eine andere Modalität, sind jedoch ebenfalls als eigenständige, natürliche Sprachen zu betrachten. Das wohldefinierte Zeichensystem genügt dabei zum Ausdruck aller pragmatischen Funktionen [Lin12].

Zur Erstellung stochastischer Modelle wird eine große semantisch annotierte Sprachdatenmenge benötigt. Bei der Entwicklung neuer Anwendungen oder für Minderheitensprachen, wie z. B. Gebärdensprachen, stehen allerdings keine geeigneten Lernstichproben zur Verfügung. Dieses „sparse data problem“ wird ebenfalls in [WL13] aufgegriffen. Unter Zuhilfenahme einer Sammlung geeigneter Sprachäußerungen, z. B. durch *Wizard-of-Oz Experimente* ermittelt, wird eine halbautomatische Konstruktion von Grammatiken vorgeschlagen.

Vorgehensweise

Anhand einer Menge von Sprachäußerungen zu einer konkreten Bedeutung werden zunächst sogenannte *Elementargrammatiken* [WL13] erzeugt. Die verwendeten Sprachäußerungen für die deutsche Lautsprache sind dabei direkt verschriftlicht worden. Zudem wurden transkribierte Videoaufnahmen von Gebärdensprachäußerungen genutzt. Diese sind von der *Fachgruppe Gebärdensprachdolmetschen* an der *Westfälischen Hochschule Zwickau* erstellt worden. Für die Symbolverarbeitung mit endlichen Automaten wird eine sequenzielle Repräsentation von Gebärdensprachzeichen benötigt. Dafür werden spezielle Glossennotationen, wie z. B. in [Ste12], verwendet. Auf diese Weise ist es möglich, dass Wort- und Glossenfolgen mit derselben Technologie verarbeitet werden können.

Durch den gezielten Einsatz von Platzhaltern in den modellierten *Elementargrammatiken* werden *Äußerungs-Bedeutungs-Transduktoren* erzeugt, welche einem *semantischen Schema* [WL13] zugehörig sind. Für eine bedeutungsbasierte Übersetzung ist die Modellierung und Verwendung eines gemeinsamen *semantischen Schemas* für Laut- und Gebärdensprachäußerungen notwendig. Die in der Applikation eingesetzten *Äußerungs-Bedeutungs-Transduktoren* [WHK⁺12, WW14] generieren semantische Zeichenketten, die zu einer *Merkmal-Werte-Relation* gehören. Die erzeugte hierarchische Bedeutungsrepräsentation muss entsprechend für die Artikulation mit dem inversen *Äußerungs-Bedeutungs-Transduktor* wieder linearisiert werden. Auf diesem Weg kann zumindest anhand von Grammatiken zu einem *semantischen Schema* gezeigt werden, dass endliche Transduktoren erfolgreich für die bidirektionale Übersetzung zwischen der syntaktischen und semantischen Ebene eines intelligenten Signalverarbeitungssystems eingesetzt werden können.

Aufbau der Arbeit

Im Anschluss an diese Einleitung erfolgt in *Kapitel zwei* die Betrachtung grundlegender Aspekte zur Sprachkommunikation und Sprachmodellierung. Aufgrund der Verwendung von Gebärdensprachäußerungen in den zu erstellenden Grammatiken erfolgt in diesem Abschnitt auch eine kurze Einführung in die Deutsche Gebärdensprache. Weiterhin wird die Verwendung von semantischen Konzepten zur Bedeutungsrepräsentation thematisiert.

Im *dritten Kapitel* werden übliche Methoden zur semantischen Interpretation und zur Artikulation sprachlicher Äußerungen vorgestellt. Dabei kann man für beide Prozesse eine Dreiteilung feststellen. Dies sind stark linguistisch geprägten Verfahren, regelbasierte Methoden sowie korpusbasierte statistische Modelle. Die Forschungsvorhaben und auch die Ausführungen in diesem Kapitel konzentrieren sich vor allem auf den Interpretationsvorgang.

Das *vierte Kapitel* stellt das verwendete Konzept des bedeutungsorientierten Sprachmodells im Kontext intelligenter Signalverarbeitungssysteme vor. In diesem Zusammenhang werden die zur Semantikrepräsentation genutzten *Merkmal-Werte-Relationen* sowie die dazugehörigen semantischen Zeichenketten vorgestellt. Auf dieser Grundlage erfolgt im Anschluss die Schilderung des prinzipiellen Einsatzes von *Äußerungs-Bedeutungs-Transduktoren* zur Interpretation und Artikulation.

Gegenstand des *fünften Kapitels* ist die Erläuterung der Verfahrensweise zur Modellierung von *Äußerungs-Bedeutungs-Transduktoren* für Sprachäußerungen der Laut- und Gebärdensprache anhand eines *semantischen Schemas*. Dabei erfolgt eine Betrachtung des verwendeten Sprachmaterials, die Erläuterung der Funktionen und Erstellung von Platzhaltern sowie der vorgenommenen Einschränkungen und möglichen Probleme bei der Modellerstellung.

Die erzeugten *Äußerungs-Bedeutungs-Transduktoren* werden in einer erstellten Applikation zur Interpretation und Artikulation eingesetzt. Anhand einer Beispieläußerung der deutschen Lautsprache erfolgt in *Kapitel sechs* die Darstellung der erzielten Ergebnisse. Weiterhin wird zur Interpretation auch ein aktuelles Verfahren umgesetzt, welches ebenfalls auf endlichen Transduktoren basiert. Das Kapitel beinhaltet deshalb auch Ausführungen zur Modellierung dieser Transduktoren und der erzielten Resultate für dasselbe *semantische Schema*.

Abgeschlossen wird die vorliegende Arbeit in *Kapitel sieben* mit einer Zusammenfassung und einem Ausblick auf weiterführende Untersuchungen.

Kapitel 2

Sprachkommunikation

2.1 Sprache als Kommunikationsform

Der Informationsaustausch zwischen technischen und auch biologischen Systemen wird als *Kommunikation* bezeichnet. Über einen Informationskanal werden dabei Signale übertragen, welche durch dynamische Systeme (Endpunkte einer Verbindung) generiert und analysiert werden.

Die Sprache ist für uns Menschen die wichtigste und natürlichste Kommunikationsform. Sie ist Träger von semantischen Informationen und leistet einen wesentlichen Beitrag zur Entwicklung der menschlichen Gesellschaft und Kultur. Der deutsche Begriff *Sprache* ist hierbei mehrdeutig [Fel12]. Im Wesentlichen kann man jedoch zwischen dem Sprachsystem einer Einzelsprache, mit seinen Zeichen und Regeln und dem Redeakt bzw. Sprachgebrauch unterscheiden.

Das Zeichensystem einer natürlichen Sprache stellt eine wichtige Grundlage der Sprachkommunikation dar. Zudem existieren verschiedene Formen von sprachlichen Zeichensystemen, wie z. B. die Lautsprache, die Gebärdensprache, die geschriebene Sprache und die Braille-Blindenschrift. Zeichen sind im Allgemeinen jedoch nicht an Sprache gebunden, dienen aber immer der Kommunikation. Beispiele für nichtsprachliche Zeichen sind u. a. akustische Signale, Verkehrszeichen, Flaggen sowie Gesten [LNP08]. Zeichen sind also mannigfaltig und umfassen theoretisch alle Sinnesmodalitäten.

Bei der Sprachkommunikation ist die Erfassung der relevanten Eigenschaften (Merkmale und Merkmalwerte) der informationstragenden Signale notwendig. Damit ist eine Entschlüsselung der vom Sender erzeugten Nachricht möglich. Der Prozess zur Informationsaufnahme wird als *Perzeption* bezeichnet. Bei der *Perzeption* durch den Menschen oder durch ein technisches System ist eine enorme Informationsreduktion aufgrund

spezifischer Abstraktionsprozesse festzustellen. Das Verstehen eines sprachlichen Ausdrucks führt im Idealfall zur Ausführung eines beabsichtigten Verarbeitungsprozesses beim Empfänger. Die Handlung bzw. *Aktion* eines Kommunikationsteilnehmers erfolgt durch die Informationsausgabe. Diese kann als inverser Übersetzungsprozess betrachtet werden und ist in der Regel mit einer Informationsausweitung verbunden. Diese Reinterpretation, bestehend aus Artikulation und Synthese, erzeugt aus einer abstrakten Symboldarstellung eine physikalische Realisierung. Eine spezielle Kommunikationsform stellt der *Dialog* dar. Bei diesem wechselseitigen Nachrichtenaustausch sind die Rollen der Kommunikationspartner alternierend.

Je nach Untersuchungsgegenstand sind im Laufe der Jahre unterschiedliche Kommunikationsmodelle entwickelt worden. In diesen Zusammenhang ist exemplarisch das *Kommunikationsmodell nach Shannon und Weaver* [Sha48, SW49] zu nennen. Weitere Ausführungen zu dieser informationstheoretischen Betrachtung des Kommunikationsprozesses findet man beispielsweise in [Fel12].

2.2 Gebärdensprache

Gehörlose Personen verwenden zur Kommunikation untereinander in der Regel die *Gebärdensprache*. Dabei können durch die Verknüpfung visuell wahrnehmbarer Sprachelemente komplexe Sachverhalte ebenso gut wie in der gesprochenen Sprache ausgedrückt werden. Ergänzend zu der Gebärdensprache wird von Gehörlosen z. B. auch das *Fingeralphabet* zum Buchstabieren von Eigennamen, Abkürzungen oder Fremdwörtern genutzt.

Auf der Welt werden unterschiedliche nationale Gebärdensprachen verwendet. Zusätzlich gibt es innerhalb eines Landes auch regional unterschiedliche Sprachvarianten und Dialekte. In diesem Kapitel liegt der Schwerpunkt auf der Beschreibung wichtiger Eigenschaften der *Deutschen Gebärdensprache* (DGS).

2.2.1 Phonologie der Gebärdensprache

Im Vergleich zur Lautsprache begann die Gebärdensprachforschung relativ spät [Dot01]. In den 1960er Jahren war es WILLIAM C. STOKOE, der die sublexikalischen Elemente der Gebärdensprache als Erster genauer untersuchte. Der von STOKOE geprägte Begriff der *Cherologie*, also die *Lehre von den Händen*, konnte sich nicht lange durchsetzen. Heute verwendet man auch hier den lautsprach-linguistischen Terminus *Phonologie*. Im Laufe der Zeit sind verschiedene Beschreibungsmodelle für gebärdensprachliche Zeichen, wie z. B. das *Bewegungs-/Halt-Modell*, *Handschicht-Modell*, *Moren-Modell* und das *Parametermodell*, entwickelt worden. Weiterführende Informationen zu diesen Modellen

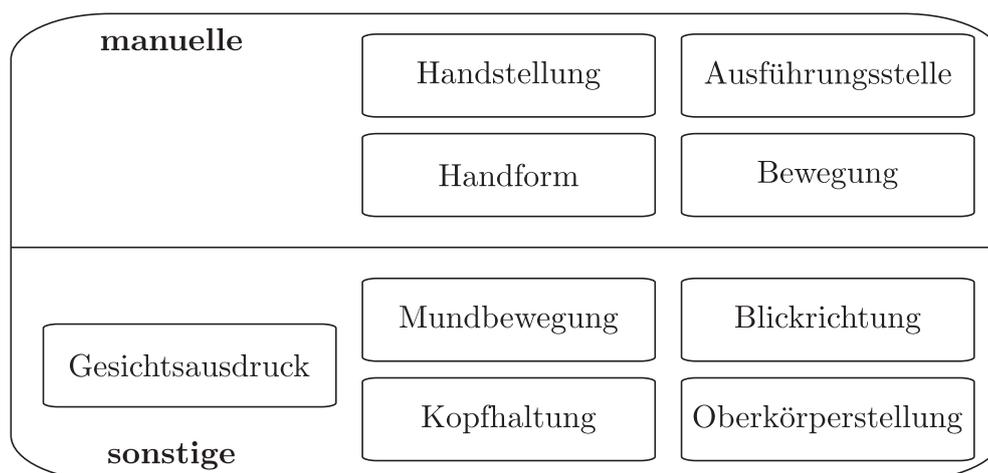


Abb. 2.1: Manuelle und nichtmanuelle Gebärdensprachparameter

findet man z. B. in [DH95, Dot01, SLM06, EHH12]. In der Hierarchie der Sprachzeichen sind Gebärden vergleichbar mit Wörtern der Lautsprache. Zur Beschreibung der Gebärdensprachzeichen soll hier nur das traditionelle *Parametermodell* betrachtet werden. Gebärden werden ein- oder beidhändig artikuliert und parallel durch bedeutungstragende nichtmanuelle und orale Elemente ergänzt. Den in Abb. 2.1 dargestellten Parametern können unterschiedliche distinktive Merkmale (Phoneme) zugeordnet werden. In der *Deutschen Gebärdensprache* existieren z. B. etwa 30 Phoneme für die Handform sowie einige bedeutungsunterscheidende Ausführungsstellen im Gebärdensraum [PMMH08]. Die simultan ausgeführten nichtmanuellen Gebärdenelemente dienen u. a. zur Kennzeichnung von Satztypen, zur Bedeutungsunterscheidung und -präzisierung sowie zur Artikulation von nicht durch Gebärden darstellbaren Adjektiven und Adverbien. Dabei existieren beispielsweise für den Gesichtsausdruck streng definierte Muster zur Vermittlung linguistischer Informationen [BB90]. Bei den Mundbewegungen muss zwischen dem *Mundgestik* und der *Mundgestik* unterschieden werden. Die *Mundgestik* unterstützt oft die Bedeutung der Gebärde. Als Beispiel ist das „ph“ bei der Gebärde SCHLAGEN zu nennen [Heß01]. Mit *Mundbild* bzw. *Ablesewort* wird hingegen die lautlose Artikulation eines Wortes aus der Lautsprache bezeichnet. Mundbilder, die simultan zur Gebärde artikuliert werden, sind durch den Kontakt mit der lautsprachlichen Welt entstanden. Einige Gebärdensprachlinguisten betrachten Ablesewörter nicht als Bestandteil einer Gebärde [EHH12].

2.2.2 Grammatische Besonderheiten der Deutschen Gebärdensprache

Eine Gebärdensprachäußerung besteht aus einer Folge von einzelnen Gebärden. Im Gegensatz zur Lautsprache erlaubt die visuell-gestische Gebärdensprache eine simultane Erzeugung und Perzeption von Phonemen. Dieses Prinzip der Simultanität kann sogar auf Lexemebene festgestellt werden. Als Beispiel für die Vereinigung von Morphemen (*Inkorporation*) kann der Ausdruck *VIER-STUNDE* angegeben werden. Hierbei wird die Zahlgebärde mit einer Zeitangabe kombiniert. Zudem tritt auch eine lineare Zusammensetzung von meist zwei Morphemen auf. Dabei werden Silbenbestandteile oder sogar ganze Silben der Gebärden weggelassen. Komposita sind beispielsweise *SCHÄFER+HUND*, *WOHN+ZIMMER* sowie *GOTT+ERWARTEN* (Advent). Wie aus Beispiel *GOTT+ERWARTEN* (Advent) aus [HV06] zu entnehmen ist, entsprechen Komposita der DGS nicht immer auch zusammengesetzten Wörtern der deutschen Lautsprache. Lexikalische Ausdrücke für Kopulae (z. B. *sein* und *haben*) sowie für unbestimmte Artikel gibt es in der DGS nicht.

Neben der Verwendung von konventionellen Gebärden können in der *Deutschen Gebärdensprache* auch spontan neue, ikonische Sprachzeichen erzeugt werden. Solche *produktiven Gebärden* werden zum Beispiel zur Beschreibung von Formen, Bewegungsabläufen und räumlichen Verhältnissen verwendet.

Viele Gebärden können bezüglich ihrer Bewegungsausführung oder ihrer Position und Orientierung im Gebärdenraum modifiziert werden. Durch eine *Verortung* werden Referenten bestimmte Raumpunkte zugeordnet. Im Verlauf der Rede kann durch eine Zeigegeste oder eine entsprechende Blickrichtung und Kopfhaltung auf ein verortetes Objekt verwiesen werden. Auf diese Weise ist es auch möglich, Beziehungen zwischen Referenten zu beschreiben.

Die Zuordnung einer Gebärde zu einer Wortart ist oft nicht durch die Form der Gebärde ermittelbar. Sie kann allerdings durch die Position im Satz oder anhand der Flexion bestimmt werden. Eine grammatische Funktion zur Angabe des Genus gibt es in der *Deutschen Gebärdensprache* nicht. So sind z. B. die 3. Person Singular und auch Artikel geschlechtsneutral. Eine Flexion von Gebärden erfolgt zum Beispiel bei *Kongruenz-* und *Raumverben*. *Kongruenzverben* beschreiben transitive Handlungen. Beispielsweise kann bei dem Verb *GEBEN* eine Zuordnung von Subjekt und Objekt durch die Anfangs- und Endposition der Bewegung im Gebärdenraum erfolgen. Weiterhin kommen Verben auch in *Klassifikatorgebärden* vor. Bei diesen komplexen Sprachzeichen kann die Hand als Stellvertreter einer Entität dienen (*Substitutor-Verben*) oder die Form eines verwendeten Gegenstands darstellen (*Manipulator-Verben*).

Eine Beschreibung der Syntax basiert in der Regel auf Basis einer Satzstruktur. Für die Gebärdensprache gibt es allerdings keine einheitlichen Kriterien zur Definition eines

Satzbegriffs [EHH12, Ebb15]. Trotzdem können z. B. bezüglich der Wortstellung einige Unterschiede zur deutschen Lautsprache festgestellt werden. In einfachen Aussagesätzen findet man oft die syntaktische Folge *Agens*>*Patiens*>*Verb*. Auch andere Wortstellungsvarianten, bei denen z. B. das Verb zwischen Agens und Patiens auftritt, sind möglich. Einen Einfluss auf die Syntax hat neben der Satzart zum Beispiel auch die *Topikalisierung*. Diese kann auch in den Strukturen der deutschen Lautsprache gefunden werden. Wichtige zu betonende Informationen werden hierbei in Richtung Satzanfang gestellt [EHH12]. Abfolgebeschreibungen werden in der DGS in der Regel chronologisch dargestellt. Manuell ausgedrückte Adverbien, wie Zeit- und Ortsangaben (z. B. *HIER* oder *MORGEN*), stehen meist am Satzanfang. Weiterhin werden große Objekte vor kleinen sowie unbewegliche Objekte vor beweglichen artikuliert. Determinative stehen zudem hinter dem Substantiv [HV06]. Dieser Abschnitt kann lediglich einen kurzen Einblick in die Morphologie und Syntax der Gebärdensprache geben. Als weiterführende Literatur kann u. a. [BB90, HV06, PMMH08, EHH12] empfohlen werden.

2.2.3 Gebärdensprach-Notationssysteme

Für die symbolische Verarbeitung gesprochener Sprache sind geeignete Repräsentationssysteme notwendig. Für die Gebärdensprachen existieren je nach Einsatzzweck verschiedene Notationssysteme. Basierend auf dem Parametermodell (vgl. Kapitel 2.2.1) gibt es phonetische Notationsformen, wie die *Stokoe-Notation* für die *American Sign Language* oder das *Hamburger Notationssystem* (HamNoSys). Weiterhin existieren einige animationsorientierte Auszeichnungssprachen zur Ansteuerung von Gebärdensprachavataren.

Ein häufig verwendetes Schriftsystem ist die *Glossennotation*. *Glossen* werden auch in dieser Arbeit zur Verschriftlichung von Gebärden verwendet. Der Glossenname repräsentiert dabei die Bedeutung eines Gebärdensprachzeichens. Unter einer *Glosse* ist eine Menge bedeutungsgleicher Gebärden zusammengefasst. Glossennamen werden mit lateinischen Großbuchstaben notiert. Das verwendete Sprachzeichen ist dabei an die jeweilige nationale Lautsprache angelehnt. Das Etikett einer Gebärde kann durch zusätzliche Symbole für grammatische Informationen ergänzt werden. Dadurch können beispielsweise Angaben für Positionen im Gebärdenraum, Betonungsinformationen oder eine mögliche Pluralbildung kodiert werden. Die *Glossenumschrift* ist auch für Nichtgebärdensprachler gut verständlich und leicht zu erlernen. Allerdings existiert für die *Glossennotation* keine einheitliche Kodierungsvorschrift. Für die Sprachverarbeitung mit endlichen Automaten ist eine *Glossenumschrift* in linearisierter Form nötig. So werden z. B. in [Ste06, Ste12] geeignete *Glossennotationen* aufgeführt. Diese basieren auf der *Aachener Glossenumschrift* [DES04] und werden z. B. für die maschinelle Übersetzung verwendet.

2.3 Zur Bedeutungsrepräsentation in sprachverarbeitenden Systemen

Das Ziel der automatischen Interpretation ist die Konstruktion einer semantischen Repräsentation zu der analysierten Sprachäußerung mit Hilfe von Wissensmodellen. Dazu muss das System (= der Interpret) eine mögliche Assoziation zwischen dem sprachlichen Ausdruck und einem Bedeutungsträger situationsabhängig selbst herstellen. In Abhängigkeit vom Anwendungsgebiet sind mögliche semantische Kategorien bereits bei der Systemmodellierung definiert worden. In Sprachdialogsystemen werden in der Regel nur semantische Einheiten in die Bedeutungsrepräsentation aufgenommen, welche für die weitere Semantikverarbeitung relevant sind.

Das vom Sprecher „Gemeinte“ ist nicht direkt beobachtbar. Mögliche Bedeutungshypothesen müssen deshalb aus dem wahrnehmbaren Sprachsignal abgeleitet und dargestellt werden. Eine Sprachzeichenfolge, wie sie als Zwischenergebnis bei der Perzeption vorliegt, ist keine adäquate Semantikepräsentation. Neben der Ambiguität solcher Folgen lassen sich auch mehrere Sprachäußerungen zu derselben Bedeutung zuordnen, was zu einer uneinheitlichen Darstellungsform führt.

Im Laufe der Zeit wurden unterschiedliche Bedeutungsrepräsentationsformen entwickelt. So werden neben formalen Logiksprachen vor allem verschiedene Möglichkeiten zur Begriffsdarstellung genutzt. Dazu gehören beispielsweise semantische Rahmen, semantische Netze sowie relationale Datenbanken. Eine weitere Möglichkeit stellen die in Kapitel 4.1 vorgestellten *Merkmal-Werte-Relationen* dar.

Ein Begriff (engl. *concept*) ist nach [Dah87] die kleinste semantische Einheit einer Wissensrepräsentation. In der ISO 1087-1 findet man folgende Definition für das Wort *concept* [ISO00]:

Definition 2.1. „*Concept* - unit of knowledge created by a unique combination of characteristics“ („characteristics - abstraction of a property of an object or of a set of objects“)

Der deutsche Begriff *Konzept* ist terminologisch nicht äquivalent zu dem englischsprachigen Begriff *concept* [Str13], wird allerdings in der deutschsprachigen Literatur synonym verwendet. Aus diesem Grund ist es notwendig, den in dieser Arbeit verwendeten Terminus *Konzept* zu definieren:

Definition 2.2. Als **Konzept** wird hier die symbolische Repräsentation von konkreten oder abstrakten Bezugsobjekten sowie von Aspekten der außersprachlichen Welt bezeichnet. Solche Informationseinheiten sind sprachenunabhängig und modalitätsneutral. Verwendete Konzeptdarstellungen umfassen eine Merkmalbezeichnung und optional einen normalisierten Datenwert.

Die getroffene Angabe zur Konzeptrepräsentation orientiert sich an [HDR⁺11]. Bedeutungsrepräsentationen zu spontansprachlichen Benutzeräußerungen werden meist in Form von flachen oder hierarchischen Konzeptstrukturen organisiert. Komplexere semantische Strukturen werden typischerweise manuell modelliert. Sie ermöglichen aufgrund eines höheren Detaillierungsgrades ein tieferes Verständnis [MBH⁺08]. Zudem sind auch die Strukturen menschlicher Gedanken eher hierarchisch aufgebaut. Obwohl sprachliche Zeichen in einer zeitlichen Abfolge artikuliert werden, müssen die dazugehörigen *Konzepte* meist zeitgleich betrachtet werden. So könnte man zum Beispiel zu der Wortfolge „*Peter steht dort*“ drei aufeinanderfolgende *Konzepte* angeben. Der verbalisierte Sachverhalt kann jedoch nur durch die simultane Verknüpfung dieser semantischen Primitiva adäquat repräsentiert werden. Trotzdem ist es natürlich möglich, dass Ideen vom Menschen auch über einen Zeitraum entwickelt oder verändert werden. Sie liegen somit nicht immer gleichzeitig vor. Konzeptuelle Strukturen dürfen demnach zeitliche Ordnungen nicht gänzlich ausschließen [Bie08].

Konzepte existieren also nicht unabhängig voneinander. Semantikrepräsentationen sollten deshalb Relationen zwischen semantischen Kategorien berücksichtigen [Sto93, KN06]. Relevante Beziehungen zwischen *Konzepten* müssen entsprechend repräsentiert oder zumindest deduktiv ermittelt werden können. Viele sprachverarbeitende Systeme verwenden jedoch eine flache Semantikstruktur [WDA05]. Ein Grund dafür ist der notwendige Aufwand bei der Erstellung der semantischen Annotation von Sprachdaten, wie sie für statistische Verfahren notwendig sind. Die Verwendung komplexer Bedeutungsrepräsentationsformen ist zeitaufwändiger und fehleranfälliger. Aus diesem Grund werden bei der Interpretation von gesprochener Sprache oft keine Relationen zwischen den Bedeutungseinheiten hergestellt. Allerdings lässt sich durch eine komplexere Attributbezeichnung ein gewisser Zusammenhang zwischen den semantischen *Konzepten* ableiten.

Nach [WDA05] können alle Repräsentationsformen von semantischen *Konzepten* als abstrakte Darstellung *semantischer Rahmen (Frame)* [Min75] aufgefasst werden. Ein *semantischer Rahmen* ist eine vorgegebene Schablone, die mit Inhalt gefüllt werden kann (vgl. Tabelle 2.1). Die freien Positionen eines *frames* werden als *slots* bezeichnet. Die Art des *slots* bestimmt den Merkmalstyp. Teilweise werden *semantische Rahmen* auch direkt aus einer Datenbankstruktur des Systems abgeleitet. *Frames* sind zur Beschreibung konkreter Situationen geeignet, in denen bestimmte Ereignisse eintreten können. Die bei der semantischen Interpretation erkannten Werte müssen der entsprechenden Attributbezeichnung zugeordnet werden (*slot-filling*). Neben Attribut-Wert-Paaren können *semantische Rahmen* auch untergeordnete Schablonen beinhalten. Auf diese Weise ist es möglich, semantische Relationen zwischen verschiedenen *Konzepten* unterschiedlicher Rahmen herzustellen. Für Sprachdialogsysteme sind solche komplexen Framestrukturen eher untypisch.

Da semantische Rahmen prinzipiell auch hierarchische Strukturen beschreiben können, lassen sich die ermittelten *Konzepte* beispielsweise als semantischer Baum darstellen.

Wortfolge: „Please show me morning flights from Boston to San Francisco on Tuesday“

Frame: SHOW:
 FLIGHTS:
 TIME:
 PART-OF-DAY: *morning*
 ORIGIN:
 CITY: *Boston*
 DEST:
 CITY: *San Francisco*
 DATE:
 DAY-OF-WEEK: *Tuesday*

Tabelle 2.1: Semantischer Rahmen zu einer gegebenen Wortfolge des ATIS-Korpus. Die eingetragenen Werte entsprechen hier den Sprachzeichen der Wortfolge. Entnommen aus [MBIS94].

Semantische Rahmen eignen sich demnach für eine sprachenunabhängige semantische Informationsrepräsentation. Sie sind allerdings stereotypische Bedeutungsrepräsentationen. Eine Berücksichtigung von unsicheren Informationen sowie die Manipulation solcher *semantischer Rahmen* ist nur eingeschränkt möglich.

2.4 Sprachmodelle

Zur Verarbeitung von Sprachzeichenfolgen einer Sprache ist es notwendig, dass ein technisches System linguistisches A-priori-Wissen besitzt. Dazu gehören verschiedene Wissensmodelle, wie z. B. die Kenntnis über die verwendeten Sprachzeichen (das Vokabular), die Wort- und Satzgrammatik und die Häufigkeit verwendeter Sprachzeichen. Das Wissen zu einer Sprache wird entweder durch Experten (*regelbasierte Sprachmodelle*) oder mit Hilfe einer Sammlung von Sprachdaten (*statistische Sprachmodelle*) für das System zur Verfügung gestellt. Diese Sprachmodelle werden in dieser Arbeit durch *endliche Automaten* repräsentiert. Nachfolgende Unterkapitel beschreiben dazu kurz die wesentlichen Aspekte.

2.4.1 Formale Grammatiken

Formale Grammatiken ermöglichen die Beschreibung und Analyse von formalen und natürlichen Sprachen. In dieser Arbeit werden die nachfolgend aufgeführten allgemeinen Notationen verwendet:

X	das Alphabet, eine endliche, nichtleere Menge von Zeichen bzw. Symbolen x_i . Das Alphabet kann auch das leere Zeichen $X^0 = \varepsilon$ enthalten.
X^*	die Menge aller Wörter über X (Kleenesche Hülle).
X^+	die Menge aller über X möglichen Wörter ohne das leere Wort. $X^+ = X^* \setminus \{\varepsilon\}$
\mathbf{x}	eine endliche Zeichenfolge über dem Alphabet X . $\mathbf{x} = (x_1, x_2, \dots, x_L)$ mit $x_i \in X$
$u \rightarrow_G v$	eine Produktionsregel der Grammatik G , welche die Symbolfolge u durch die Zeichenfolge v ersetzt.
$ \mathbf{x} = L$	die Länge der Zeichenfolge \mathbf{x} .
\mathcal{L}	eine formale Sprache über X , mit $\mathcal{L} \subseteq X^*$.

Für die weiteren Betrachtungen beschränken wir uns auf die formalen Grammatiken nach CHOMSKY. NOAM CHOMSKY beschreibt durch seine erzeugenden Chomsky-Grammatiken G entsprechende Systeme, mit denen durch Anwendung einer Menge von Produktionsregeln \mathcal{P} eine (formale) Sprache $\mathcal{L}(G)$ generiert werden kann.

Definition 2.3. Eine (*generative*) *Grammatik* G ist allgemein ein *Quadrupel* (N, X, \mathcal{P}, S) mit

- der endlichen Menge von Nichtterminalen N ,
- der endlichen Menge von Terminalen X , wobei $N \cap X = \emptyset$ gilt
- einer endlichen Menge von Produktionsregeln \mathcal{P} ,
mit $\mathcal{P} \subset \{u \rightarrow_G v \mid u \in (N \cup X)^+, v \in (N \cup X)^*\}$
- dem Startsymbol $S \in N$ [PK08].

Die Produktionsregeln definieren, welches u in der Grammatik G durch v ersetzt werden kann. Für die Ableitungsschritte ist sichergestellt, dass auf der linken Seite mindestens ein Nichtterminal (Hilfssymbol) steht. Die rechte Seite kann hingegen auch nur das leere Symbol $v = \varepsilon$ beinhalten.

Definition 2.4. $\mathcal{L}(G) := \{\mathbf{x} \in X^* \mid S \Rightarrow^* \mathbf{x}\}$ ist die von einer Chomsky-Grammatik G generierte Sprache.

Dabei enthält die Sprache $\mathcal{L}(G)$ die Menge der Wörter, welche durch die Ableitungen $S \Rightarrow^* \mathbf{x}$ (wiederholte Anwendung von Produktionsregeln) von G erzeugt werden können. Formale Grammatiken können auch zur Analyse von natürlichen Sprachen verwendet werden. Die Terminalsymbolmenge einer Sprachäußerungsgrammatik entspricht dann der Menge aller Wörter der natürlichen Sprache \mathcal{L} . Zu den Chomsky-Grammatiken

gehören 4 verschiedene Typen. Die allgemeine Chomsky-Grammatik entspricht hier dem Typ 0. Die am stärksten eingeschränkte formale Grammatik ist die *reguläre Grammatik* G_3 bzw. Typ-3-Grammatik. Die linke Seite einer Produktionsregel muss genau ein Nichtterminal aufweisen. Auf der rechten Seite darf ein Terminal und maximal ein weiteres Nichtterminal stehen. Wenn die Produktionsregeln auf der rechten Seite zwei Symbole aufweisen, dann wird entweder immer das Terminalsymbol oder immer das Nichtterminalsymbol zuerst aufgeführt.

Zu jeder *regulären Grammatik* G_3 kann ein äquivalenter *endlicher Automat* (siehe Kapitel 2.4.2) angegeben werden. Die von einem endlichen Automaten akzeptierte Zeichenfolge \mathbf{x} gehört zu der regulären Sprache $\mathcal{L}(G_3)$. Die Sequenzlänge $|\mathbf{x}|$ ergibt sich in der Regel aus der Anzahl der Zustandsübergänge (Anwendungen von Produktionsregeln) zur Ableitung von \mathbf{x} . Hierfür dürfen allerdings Zustandsübergänge mit dem leeren Symbol ($|\varepsilon| = 0$) nicht berücksichtigt werden. Reguläre Grammatiken können Abfolgen von Terminalsymbolen produzieren und dabei zusätzlich alternative Teilfolgen sowie beliebige Wiederholungen von Sprachzeichen ausdrücken.

2.4.2 Endliche Automaten

Ein *endlicher Automat* \mathcal{A} (*finite state machine*) ist ein mathematisches Modell, welches für die Verarbeitung von regulären Sprachen verwendet werden kann.

Algebraische Automatendefinition

Definition 2.5. *Ein gewichteter endlicher Automat \mathcal{A} über einem Gewichtshalbring $S = (\mathbb{K}, \oplus, \otimes, \bar{0}, \bar{1})$ ist algebraisch betrachtet ein 6-Tupel $\mathcal{A} = (Z, I, F, X, Y, E)$ mit*

- *der endlichen Menge aller Zustände Z ,*
- *der Menge aller Anfangszustände $I \subseteq Z$,*
- *der Menge aller Schlusszustände $F \subseteq Z$,*
- *einem endlichen Eingabealphabet X ,*
- *dem endlichen Ausgabealphabet Y und*
- *einer endlichen Multimenge von gewichteten Zustandsübergängen E [Moh09].*

Zusätzlich kann noch die initiale Abbildungsfunktion $I \rightarrow \mathbb{K}$ und die finale Gewichtungsfunktion $F \rightarrow \mathbb{K}$ angegeben werden. Die in einem gewichteten Automaten enthaltenden Zustandsübergänge E sind dabei eine Teilmenge aus $Z \times (X \cup \{\varepsilon\}) \times (Y \cup \{\varepsilon\}) \times \mathbb{K} \times Z$. Ein einzelner Zustandsübergang $e = (z, x, y, w, z') \in E$ ist ein Tupel, bestehend aus

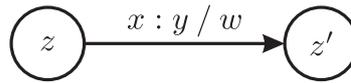


Abb. 2.2: Zustandsübergang eines gewichteten Transduktors

Startzustand $z \in Z$, Zielzustand $z' \in Z$, Eingabesymbol $x \in X$, Ausgabesymbol $y \in Y$ sowie einem von $\bar{0}$ verschiedenen Gewicht $w \in \mathbb{K}$. Eine solche Transition e ist in Abb. 2.2 visualisiert. Die Gewichte w können Wahrscheinlichkeiten oder Abstände repräsentieren und stammen aus einem Gewichtshalbring S . Sind bei einer Transition sowohl das Eingabesymbol x als auch Ausgabesymbol y gleich ε , so wird diese als *Epsilonkante* bezeichnet.

Die algebraische Struktur $S = (\mathbb{K}, \oplus, \otimes, \bar{0}, \bar{1})$ ist ein *Halbring*, mit der kommutativen bzw. abelschen Halbgruppe $(\mathbb{K}, \oplus, \bar{0})$ und der Halbgruppe $(\mathbb{K}, \otimes, \bar{1})$. Da diese Halbgruppen jeweils ein neutrales Element besitzen, werden sie auch als *Monoide* bezeichnet. Hier ist demnach $\bar{0}$ das Nullelement und $\bar{1}$ das Einselement des Halbrings S . Bezüglich der Multiplikation muss das Nullelement absorbierend sein, d. h. es gilt für alle $k \in \mathbb{K} : k \otimes \bar{0} = \bar{0} \otimes k = \bar{0}$.

Halbring	\mathbb{K}	\oplus	\otimes	$\bar{0}$	$\bar{1}$
Boolescher \sim	$\{0, 1\}$	\vee	\wedge	0	1
Wahrscheinlichkeits- \sim	$\mathbb{R}_+ \cup \{\infty\}$	+	\cdot	0	1
logarithmischer- \sim	$\mathbb{R} \cup \{-\infty, +\infty\}$	\oplus_{\ln}	+	$+\infty$	0
tropischer- \sim	$\mathbb{R} \cup \{-\infty, +\infty\}$	min	+	$+\infty$	0
Zeichenketten- \sim	$X^* \cup \{\infty, \varepsilon\}$	\wedge	\cdot	x_∞	ε

\oplus_{\ln} ist definiert als: $x \oplus_{\ln} y = -\ln(e^{-x} + e^{-y})$

x_∞ : unendliche Zeichenkette ($\forall x \in X^* \cup \{x_\infty\} : x \wedge x_\infty = x_\infty \wedge x = x$)

Tabelle 2.2: Typische Gewichtshalbringe [Moh09, HW15]

Wichtige Halbringe sind in Tabelle 2.2 dargestellt. So können beispielsweise Gewichtshalbringe mit einer Menge von Zeichenketten beschrieben werden. Dabei liefert die binäre \oplus -Operation das längste gemeinsame Präfix (Beispiel: $aber \wedge abend = abe$) und die \otimes -Operation die Verkettung von Zeichen. Weiterhin kann ein ungewichteter endlicher Automat auch als gewichteter Automat über den booleschen Halbring betrachtet werden [Moh09, HW15].

Automatengraphen

Graphen sind Mengen, in denen die Elemente in bestimmten Relationen zueinander stehen. Ein Graph $G = (V, E)$ besteht aus einer Menge von Knoten V (vertices) und einer

Kantenmenge E (edges), wobei eine Kante zwei Knoten miteinander verbindet. Endliche Automaten lassen sich als *gerichtete*, etikettierte Graphen darstellen. Jede Kante $e = \rightarrow = (v_{i-1}, v_i)$ ist ein geordnetes Paar und besitzt eine Orientierung. Ein gerichteter Graph G ist somit ein Paar (V, \rightarrow) mit $\rightarrow \subseteq V \times V$. Die Orientierung wird üblicherweise durch Pfeile an den Kanten dargestellt. Die Etikettierung der Kanten ergibt sich durch die verwendeten Eingabezeichen ($x_i \in X$) und die übersetzten Ausgabesymbole ($y_i \in Y$). Für die endlichen Automaten entspricht die Knotenmenge V der Zustandsmenge Z . Knoten werden grafisch durch Kreise repräsentiert. Im Bild 2.3 sind diese mit $z_i \in Z$ bezeichnet.

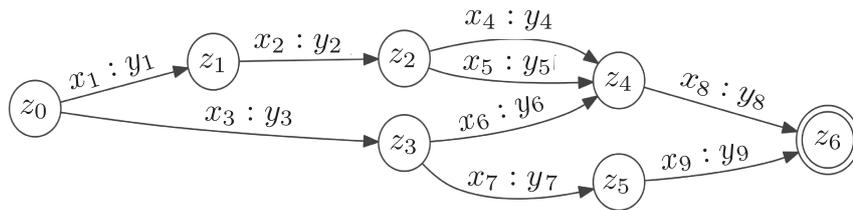


Abb. 2.3: Visualisierung eines ungewichteten Automaten als Graphen

Endliche Automaten besitzen Anfangs- und Schlusszustände aus der Menge der Knotenmenge V . Die Schlusszustände werden im Allgemeinen mit einem doppelten Kreis dargestellt. Diese können dabei gleichzeitig auch aus der Menge der Anfangszustände stammen. Oft besitzen Automaten nur einen Anfangszustand. Dieser ist in Abb. 2.3 mit z_0 beschriftet.

Die Möglichkeit der Visualisierung hat den Vorteil, dass z. B. erstellte Automaten leichter durch den Menschen verifiziert werden können. Wenn allerdings die Automatengraphen sehr komplex werden, geht die Lesbarkeit und die Übersichtlichkeit schnell verloren.

Pfade endlicher Automaten

Ein Element E^* eines endlichen Automaten \mathcal{A} mit aufeinanderfolgenden Transitionen $e_i \in E$, bei dem kein Zustand mehrmals enthalten ist, wird als *Pfad* π bezeichnet. $\mathcal{U}(I, F)$ ist die Menge aller durchgehenden (akzeptierenden) Pfade, d. h. aller Pfade von den Anfangszuständen $z_i \in I$ zu den Schlusszuständen $z_i \in F$. Ein durchgehender Pfad entspricht einer möglichen Ableitung einer zu \mathcal{A} äquivalenten regulären Grammatik G . Das Gewicht eines Pfades $w[\pi]$ wird durch

$$w[\pi] = \bigotimes_{e_i=1}^{|\pi|} w(e_i) \quad (2.1)$$

berechnet, wobei $|\pi|$ die Länge des Pfades angibt. Nachfolgend bezeichnet \mathbf{x} eine endliche Folge von Eingabesymbolen und \mathbf{y} die Ausgabesymbolsequenz. Die Zuordnung

des Gesamtgewichtes zu dem Zeichenkettenpaar $(\mathbf{x}, \mathbf{y}) \in X^* \times Y^*$ durch den Automaten \mathcal{A} wird durch die Gleichung

$$\mathcal{A}(\mathbf{x}, \mathbf{y}) = \bigoplus_{\pi \in \mathcal{U}(I, \mathbf{x}, \mathbf{y}, F)} w[\pi] \quad (2.2)$$

beschrieben. Sollten also mehrere durchgehende Pfade mit derselben Eingabesequenz \mathbf{x} und Ausgabesequenz \mathbf{y} existieren, werden die Pfadgewichte $w[\pi]$ summiert. In [Moh09] sind für die Berechnung des Gesamtgewichtes zusätzlich die initiale und finale Gewichtungsfunktion aufgeführt. Da in dieser Arbeit jedoch den Anfangs- und Schlusszuständen keine Gewichte zugeordnet werden, ist auf die Angabe dieser Gewichtungsfunktionen in der Formel 2.2 verzichtet worden.

Generatoren, Akzeptoren und Transduktoren

Anhand der Charakteristika von Ein- und Ausgabealphabeten lassen sich endliche Automaten wie folgt klassifizieren [HW15]:

- **Akzeptor**

Ein endlicher Automat mit leerem Ausgabealphabet ($Y = \emptyset$) wird als Akzeptor bezeichnet. Durchgehende Pfade $\pi \in \mathcal{U}(I, \mathbf{x}, F)$ eines Automaten heißen *akzeptierend*.

- **Generator**

Generatoren emittieren Ausgabewörter \mathbf{y} , ohne dass bei den Transitionen Eingabezeichen gelesen werden. Die Menge der Eingabesymbole ist in diesem Fall leer ($X = \emptyset$).

- **Transduktor**

Automaten welche sowohl ein nichtleeres endliches Eingabealphabet X als auch ein nichtleeres endliches Ausgabealphabet Y verwenden, werden als *endliche Transduktoren* (englisch: *finite state transducer*, FST) bezeichnet. Diese beschreiben Relationen zwischen regulären Mengen und können einer Übersetzung ein Gewicht zuordnen.

Für die nachfolgende Beschreibung von Automatenoperationen wird allgemein die Bezeichnung *Transduktor* verwendet.

Automatenoperationen

Die Verwendung von Transduktoren in sprachverarbeitenden Systemen erfordert u. a. die Möglichkeiten, Automaten zu manipulieren und zu kombinieren. Für die praktische Umsetzung der Beispielapplikation in dieser Arbeit wurden die implementierten Algorithmen der Open-Source Bibliothek *OpenFst* [ARS⁺07] verwendet. Nachfolgend werden nur die Operationen kurz vorgestellt, welche für die praktischen Untersuchungen genutzt wurden. Eine ausführliche Beschreibung dieser und weiterer Automatenoperationen findet man z. B. in [MPR08, Moh09, HW15].

- **Summe** (*OpenFst: Union*)

Die Summe zweier Transduktoren beinhaltet die Zustandsmengen Z_1 und Z_2 sowie die Ein- und Ausgabealphabete von T_1 und T_2 . Die Gewichte bleiben erhalten. Dazu wird im Open-Source-Projekt *OpenFst* der Anfangszustand eines Automaten mittels einer Epsilon-Kante mit dem Anfangszustand des zweiten Transduktors verbunden.

$$\forall (\mathbf{x}, \mathbf{y}) \in X^* \times Y^* : (T_1 \oplus T_2)(\mathbf{x}, \mathbf{y}) = T_1(\mathbf{x}, \mathbf{y}) \oplus T_2(\mathbf{x}, \mathbf{y}) \quad (2.3)$$

Die Summenoperation ist kommutativ. Die auf diesem Weg erzeugten Transduktoren sind *äquivalent*. Zwei Transduktoren werden als *äquivalent* bezeichnet, wenn sie beliebigen Zeichenkettenpaaren (\mathbf{x}, \mathbf{y}) dasselbe Gewicht zuordnen.

- **Kleenescher Abschluss** (*OpenFst: Closure*)

Der Kleenesche Abschluss X^* ermöglicht die beliebige n -fache Verkettung von allen akzeptierten Zeichenfolgen \mathbf{x} eines Automaten. Allgemein ergibt sich somit aus $T(\mathbf{x}, \mathbf{y})$ der Transduktor $T^*(\mathbf{x}, \mathbf{y})$, mit der Gleichung

$$\forall (\mathbf{x}, \mathbf{y}) \in X^* \times Y^* : T^*(\mathbf{x}, \mathbf{y}) = \bigoplus_{n=0}^{+\infty} T^n(\mathbf{x}, \mathbf{y}). \quad (2.4)$$

Dazu werden die Schlusszustände mit allen Anfangszuständen vom Transduktor T durch Epsilon-Kanten verknüpft. Zusätzlich wird ein neuer Anfangszustand eingefügt, welcher gleichzeitig auch ein Schlusszustand ist ($L^0 = \varepsilon$). Somit beinhaltet der Kleenesche Abschluss auch das leere Wort ε . Will man dieses leere Wort aus dem Ergebnis ausschließen, so muss man die positive Hülle (Kleene-+-Abschluss) verwenden. Diese ist in *OpenFst* ebenfalls als Option beim Kleeneschen Abschluss realisiert.

- **Inversion (Transposition)** (*OpenFst: Invert*)

Diese Operation erzeugt aus einem gewichteten Transduktor einen neuen FST, bei dem die Ein- und Ausgabezeichen an jedem Zustandsübergang vertauscht

sind. Die Gewichte werden bei dieser Abbildungsumkehrung nicht verändert.

$$T^{-1}(\mathbf{x}, \mathbf{y}) = T(\mathbf{y}, \mathbf{x}) \quad (2.5)$$

Durch die Inversion erfolgt ein Wechsel von der Eingabesprache L_X und der Ausgabesprache L_Y .

- **Komposition** (*OpenFst: Compose*)

Eine Nacheinanderausführung zweier Automaten wird durch diese Kombinationsoperation erreicht. Das Gewicht des Kompositionsergebnisses ergibt sich durch die \otimes -Operation

$$(T_1 \circ T_2)(\mathbf{x}, \mathbf{y}) = \bigoplus_{\mathbf{z}} T_1(\mathbf{x}, \mathbf{z}) \otimes T_2(\mathbf{z}, \mathbf{y}). \quad (2.6)$$

Wenn der Transduktor T_1 also eine Zeichenkette \mathbf{x} in die Folge \mathbf{z} übersetzt, dann generiert der Transduktor T_2 die Zeichenkette \mathbf{y} , wenn er \mathbf{z} akzeptiert.

- **Ersetzung** (*OpenFst: Replace*)

Diese Automatenoperation realisiert eine rekursive Ersetzung von Kanten eines Automaten durch einen anderen Transduktor. Dabei wird in jeden Zustandsübergang des ursprünglichen Automaten, welcher als Ausgabesymbol ein zu definierendes Nichtterminal besitzt, der zugehörigen Automat eingesetzt.

- **Kürzeste durchgehende Pfade** (*OpenFst: ShortestPath*)

Diese Suchoperation liefert als Ergebnis einen Transduktor mit den n -Pfadern, welche die geringsten Pfadkosten $w(\pi_i)$ aufweisen.

- **Entfernung von Epsilon-Transitionen** (*OpenFst: Rmepsilon*)

Dieser Algorithmus entfernt alle Epsilonanten eines Transduktors. Eine Epsilon-Transition ist ein Zustandsübergang, welcher sowohl ein leeres Eingabewort als auch ein leeres Ausgabewort besitzt. Die Optimierungsoperation erzeugt einen *äquivalenten* Transduktor $\kappa(T)$ ohne Epsilonanten. Epsilon-Transitionen zu Schlusszuständen können allerdings nicht eliminiert werden [HW15].

- **Determinierung** (*OpenFst: Determine*)

Die Determinierung $\text{det}(T)$ erzeugt zu einem nicht-deterministischen Transduktor einen äquivalenten deterministischen Automaten. Dabei ist zu beachten, dass nicht alle Automaten determinierbar sind [Moh09, HW15]. Ein Automat ist *determiniert*, wenn es nur einen Anfangszustand z_0 gibt und jeder Zustand nur ausgehende Transitionen mit unterschiedlichen Eingabesymbolen besitzt. In jedem Zustand z_i mit mehreren ausgehenden Kanten wird somit dem Paar $[z_i, x]$ genau ein Paar $[y, z'_i]$ zugeordnet. Aus diesem Grund übersetzen determinierte Automaten dieselbe Eingabesequenz auch immer in dieselbe Ausgabezeichenkette. Eine ε -Transition wird hierbei als regulärer Übergang betrachtet.

- **Minimierung** (*OpenFst: Minimize*)

Bei dieser Optimierungsoperation wird die Anzahl der Zustände verringert, in dem äquivalente Zustände zusammengelegt werden. Durch die Minimierung $\min(T)$ wird ein *äquivalenter* gewichteter Transduktor erzeugt. Bei dem in der *OpenFst*-Bibliothek verwendeten Algorithmus ist es erforderlich, dass der zu minimierende Automat deterministisch ist.

2.4.3 Statistische Sprachmodelle

Statistische Sprachmodelle sind stochastische Grammatiken und werden anhand einer Lernstichprobe trainiert. Damit ist eine Schätzung der Wahrscheinlichkeit einer analysierten bzw. übersetzten Sprachäußerungen möglich. Für eine endliche Sprachzeichenfolge \mathbf{x} wird durch die reguläre stochastische Grammatik die Verbundwahrscheinlichkeit $P(\mathbf{x})$ bestimmt. Diese kann mit den bedingten Wahrscheinlichkeiten der verwendeten Sprachzeichen x_i durch

$$\begin{aligned} P(\mathbf{x}) &= P(x_1) \cdot P(x_2|x_1) \cdot P(x_3|x_1x_2) \cdot \dots \cdot P(x_L|x_1 \dots x_{L-1}) \\ &= \prod_{i=1}^L P(x_i|x_1 \dots x_{i-1}) \end{aligned} \quad (2.7)$$

bestimmt werden. Typischerweise werden *N-Gramm-Sprachmodelle* als endliche Repräsentation einer Sprache \mathcal{L} verwendet. Endliche Automaten bieten die Möglichkeit einer kompakten Repräsentation eines *N-Gramm-Sprachmodells* [MPR08]. Grundlage für diese Modellvorstellung ist die Annahme, dass die jüngere Vergangenheit einen größeren Einfluss auf die Vorhersage des nächsten Sprachzeichens hat. Deshalb verwendet man für die Bewertung von dem Wort x_i anstelle der gesamten Folge ab dem 1. Wort nur die $N - 1$ vorherigen Sprachzeichen. Die approximierte Wahrscheinlichkeit ergibt sich mit

$$P(\mathbf{x}) \approx \prod_{i=1}^L P(x_i|x_{i-(N-1)} \dots x_{i-1}) \quad (2.8)$$

Je kleiner dabei N ist, umso größer fällt die Approximation aus. In einem Sprachverarbeitungssystem können auch fehlerbehaftete Symbolfolgen und somit auch unbeobachtete N-Gramme auftreten, deren geschätzte Wahrscheinlichkeit Null ist. Diese Problematik wird zudem mit zunehmender Kontextlänge größer. Aus diesem Grund erfolgt eine *Glättung* (Smoothing) der relativen N-Gramm-Häufigkeiten oder die Erstellung kombinierter N-Gramm-Modelle (*Rückfall* und *Interpolation*). Weitere Einzelheiten zu statistischen Sprachmodellen findet man u. a. in [ST95, CG99, Fin14, JM14].

Kapitel 3

Verfahren zur Interpretation von gesprochener Sprache und zur Generierung von Wortfolgen

Dieses Kapitel gibt einen Überblick zu möglichen Verfahren der semantischen Interpretation von erkannten Wortfolgen sowie zur Generierung von natürlichsprachlichen Äußerungen aus einer Bedeutungsrepräsentation. Ziel ist die Vermittlung der grundlegenden Prinzipien der verwendeten Modelle.

3.1 Semantische Interpretation von gesprochener Sprache

Ziel der semantischen Interpretation ist die automatische Generierung einer Bedeutungsrepräsentation zu der analysierten Sprachzeichenfolge. Im Laufe der Zeit sind dafür verschiedene Verfahren entwickelt worden. Heutige Sprachdialogsysteme verwenden meist eine regelbasierte Analyse mit semantischen Grammatiken [TWH13]. Dabei werden meist nur die Phrasen oder Wörter einer erkannten natürlichsprachlichen Benutzeräußerung identifiziert, welche für den jeweiligen Dialogschritt relevant sind. Aus diesen Wörtern werden dann die domänenspezifischen semantischen *Konzepte* abgeleitet. Die ermittelte Bedeutungsrepräsentation zu einer kompletten Benutzeräußerung ist in den meisten Fällen relativ einfach und flach strukturiert [Mac09].

Die gestiegenen rechentechnischen Möglichkeiten zur Verarbeitung größerer Datenmengen sowie die erfolgreiche Etablierung statistischer Methoden bei der Spracherkennung waren vermutlich ausschlaggebend für den Einzug stochastischer Modelle auf dem gesam-

ten Gebiet der sprachverarbeitenden Systeme. Für die zu schätzenden Modellparameter werden allerdings umfangreiche semantisch aufbereitete Sprachdaten zu der jeweiligen Domäne benötigt. Aus diesem Grund stellen bestehende semantisch beschriftete Datenbasen eine wichtige Grundlage für technologische Entwicklungen dar. Beispiele hierfür sind der *ATIS-Korpus* (Air Travel Information System), von dem Teile auch in die französische Sprache übersetzt wurden (L'ATIS) sowie der französischsprachige *MEDIA-Korpus* [BMQD09] aus dem MEDIA-EVALDA Projekt.

Es gibt auch Aufgabenbereiche die mit der semantischen Interpretation eng verwandt sind, wie u. a. die Informationsextraktion und die thematische Zuordnung von Wortfolgen (siehe Tabelle 3.1). Ein Beispiel hierfür ist die stochastische Klassifikation der

Wortfolge	<i>Ich</i>	<i>möchte</i>	<i>morgen</i>	<i>von</i>	<i>Berlin</i>	<i>nach</i>	<i>Dresden</i>	<i>fahren</i>
Konzeptbeschriftung	O	O	date_rel-B	O	arr-B	O	dept-B	O
Entitätserkennung	O	O	O	O	city-B	O	city-B	O
Intention	find_rail_connection							
Domäne	railway_information							

Tabelle 3.1: Typische Aufgaben der semantischen Sprachanalyse nach [MHDB13]. Die semantische Beschriftung erfolgt hier mit der typischen IOB-Annotation [RM95].

Intention eines Telefongesprächs in [GRW97]. Auf diese Weise kann der Anrufende direkt an den zuständigen Mitarbeiter weitergeleitet werden. Als weiteres Beispiel kann in diesem Zusammenhang die Segmentierung und Klassifizierung von Sprachdialogen durch das CALO-System [TSV⁺10] genannt werden.

Die allgemeine Zuordnung des Dialogakttyps, Themas bzw. der Domäne zu einer kompletten Benutzeräußerung soll nachfolgend nicht näher betrachtet werden. Erfolgt in einem intelligenten System allerdings eine Klassifikation des Dialogakts, so können diese Informationen z. B. bei einer Entscheidungsfindung oder auch zur Semantikrepräsentation genutzt werden.

Historisches

Vor allem seit den 1970er Jahren werden entsprechende Systeme für die Interpretation gesprochener Sprache entwickelt. Diese beruhen typischerweise auf dem Prinzip der syntaxbasierten semantischen Interpretation (siehe Kapitel 3.1.1) [Mor07]. Hierzu wurden u. a. auch erweiterte Übergangnetzwerke (Augmented Transition Networks (ATN)) [Woo70, CM79] verwendet.

Eine der ersten Veröffentlichungen, welche stochastische Grammatiken zur Verarbeitung gesprochener Sprache verwendete, ist das System *TINA* [Sen92] vom Massachusetts Institute of Technology (MIT). Der Parser basiert auf einer probabilistischen kontextfreien

Grammatik. Dabei werden sowohl syntaktische als auch semantische Informationen verarbeitet.

Hierarchische Merkmalstrukturen können von dem *Hidden Understanding Model* [MSBI94] erzeugt werden. Dieses generative Verfahren ist als rekursives Übergangsnetzwerk realisiert worden. Für jedes nichtterminale Merkmal wird ein eigener stochastischer Automat modelliert, wobei die enthaltenden Zustände abstrakten oder individuellen Merkmalen entsprechen. Diese Netzwerke bilden zusammen ein semantisches Sprachmodell. Weiterhin sind in dem stochastischen Netzwerk die Übergangswahrscheinlichkeiten zwischen Wörtern in Abhängigkeit von terminalen Merkmalbezeichnungen angegeben. Die Dekodierung erfolgt durch die Suche nach dem wahrscheinlichsten Pfad des kombinierten Netzwerks zu der gegebenen Sprachzeichenfolge.

In dem *CHRONUS*-System [LP95] wird als Übersetzungsmodell ein probabilistisches Netzwerk genutzt. Der verwendete Ansatz ist vergleichbar mit einem Hidden-Markov-Modell, wobei die Observationen Wörter sind. Versteckte Zustände des Modells entsprechen den semantischen *Konzepten*. Durch die Viterbi-Dekodierung wird die Zuordnung von Konzeptbeschriftungen zu der gegebenen Sprachäußerung erreicht. Durch dieses System wird eine flache Bedeutungsstruktur (Schablonen mit Attribut-Wert-Paaren) erzeugt.

3.1.1 Syntaxbasierte semantische Interpretation

Ansatz dieser Verfahren ist die sprachliche Determinierung der Bedeutung einer Äußerung. Grundlage für die Bedeutungsbestimmung auf linguistischer Ebene ist zumeist das *Kompositionalitätsprinzip*, welches auf Gottlob Frege zurückgeht. Dieses Prinzip wird vor allem von Vertretern der strukturalistischen und logischen Semantik propagiert [Klo11]. Ausführliche Betrachtungen zum Thema Kompositionalität findet man z. B. in [HMW12]. Die kompositionelle Semantik eines Satzes ergibt sich demnach aus den einzelnen Wortbedeutungen, unter der Berücksichtigung der syntaktischen Beziehungen zwischen lexikalischen Einheiten. Zur Generierung einer Semantikrepräsentation wird eine grammatische Analyse der Sprachzeichenfolge durchgeführt (vgl. Abb. 3.1). Die Modellierung von semantischem Wissen hängt in der Praxis von dem jeweiligen Anwendungsgebiet ab. Eine Trennung von syntaktischer und semantischer Analyse bietet allerdings den Vorteil einer höheren Anwendungsunabhängigkeit.

Ein zu untersuchender Satz besteht aus einer geordneten Menge an Phrasen. Die morpho-syntaktische Analyse des sprachlichen Ausdrucks liefert dessen grammatische Konstituenten. Für das Parsing ist eine geeignete Grammatik der zu analysierenden Sprache notwendig. Bei der Klassifikation liefert der Parser zu einer grammatisch korrekten

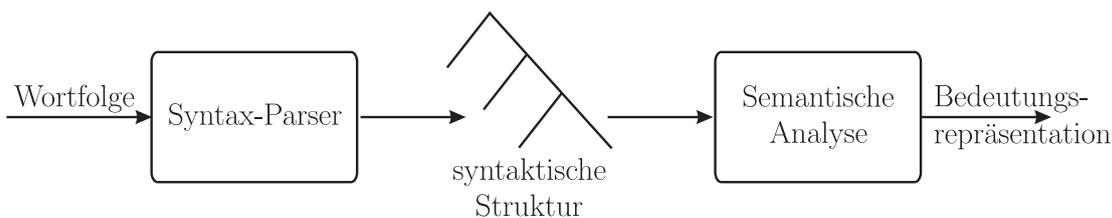


Abb. 3.1: Grundprinzip der syntaxgesteuerten semantischen Analyse nach [JM00]

Sprachzeichenfolge eine syntaktische Repräsentation (Tiefenstruktur). Die Syntaxrepräsentation kann z. B. ein Syntaxbaum (Ableitungsgraph) sein. Für die syntaxgesteuerte semantische Analyse können auch stochastische Grammatiken bzw. Netzwerke verwendet werden. Dabei werden den Produktionsregeln anhand der Auftretenshäufigkeit in der Datenbasis entsprechende Wahrscheinlichkeitswerte zugeordnet.

Die entstandenen Syntaxstrukturen werden bei der semantischen Analyse, unter Berücksichtigung von lexikalischem Wissen, regelbasiert interpretiert. Ziel dabei ist die Disambiguierung der lexikalischen Bedeutungen und die Überführung in eine Bedeutungsrepräsentation. Formal-logische Ausdrücke, semantische Rahmen oder auch semantische Netze sind mögliche symbolische Repräsentationen.

Teilweise wird die Syntaxstruktur durch entsprechende Ableitungsregeln direkt mit semantischen Beschriftungen angereichert. Dieser Vorgang wird als *Semantic Role Labeling* bezeichnet. Beschriftungen können spezielle domänenspezifische Merkmale oder auch allgemeine semantische Markierungen, wie die z. B. die semantischen Rollen *Agens* (Akteur) und *Patiens* (dem von der Handlung Betroffenen) sein. Weiterführende Information dazu findet man zum Beispiel in [GJ02, CM05, MCLS08]. In der Abb. 3.2 ist ein möglicher Ableitungsbaum mit semantischen Rollen zu dem analysierten Satz „Peter trinkt das Bier“ dargestellt.

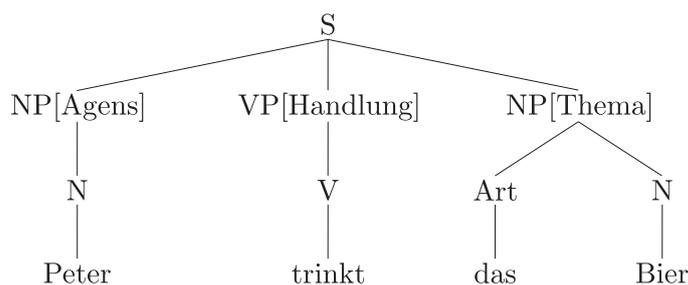


Abb. 3.2: Syntaxbaum mit semantischer Etikettierung nach [MBH+08]

Probleme der syntaxbasierten semantischen Analyse

Wie schon in [Lin15] beschrieben, sind die Nutzung der grammatischen Informationen aus der Analyse von spontansprachlichen Äußerungen und die Fokussierung auf die Satzstruktur oft nicht zielführend. Die vom Menschen verwendete Spontansprache weist nämlich u. a.

- eine andere Grammatik, im Vergleich zur geschriebenen oder vorgelesenen Sprache,
- Wiederholungen, Selbstkorrekturen und Häsitationen durch den Sprecher und
- unvollständige Sätze (Satzabbrüche)

auf. Zudem berücksichtigt die wörtliche Interpretation in der Regel nicht den bestehenden Interaktionskontext. Diese Besonderheiten müssen allerdings bei der semantischen Interpretation in Sprachdialogsystemen berücksichtigt werden. Aus dieser Perspektive heraus betrachtet ist eine syntaxbasierte semantische Interpretation von frei formulierten Benutzeräußerungen für viele praktische Anwendungen ungeeignet.

3.1.2 Wissensbasierte Interpretation mit semantischen Grammatiken

Semantische Grammatiken dienen zur direkten Bedeutungsanalyse einer Oberflächenstruktur. Sie werden z. B. zur schematischen Erfassung relevanter semantischer Informationen in Sprachdialogsystemen genutzt. Bei der Erstellung solcher konzeptuellen Grammatiken muss für eine korrekte Konzeptzuordnung die Variabilität von spontansprachlichen Äußerungen berücksichtigt werden. Im Prinzip erreicht man dies durch eine Detektion von Schlüsselwörtern bzw. semantisch relevanten Phrasen. Wenn die Dialoginitiative vorwiegend beim System liegt, lassen sich auch auf diese Weise zuverlässige Systeme realisieren. Als Ergebnis erhält man zu der interpretierten Wortfolge eine Menge von semantischen *Konzepten*.

Ein Minimalbeispiel einer semantisch geparsten Wortfolge \mathbf{x} ist in Abb. 3.3 dargestellt.

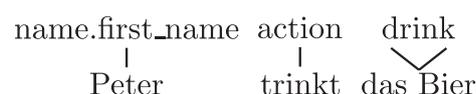


Abb. 3.3: Beispiel eines semantischen Ableitungsgraphen nach [BSH08]

Hierbei werden die Attribute a_i dem Ausdruck $\mathbf{x} = \text{„Peter trinkt das Bier“}$ direkt zugeordnet. Die dazugehörigen Werte des *Konzepts* ergeben sich hierbei aus dem zugeordneten Wort.

Im Gegensatz zur *syntaxbasierten semantischen Interpretation* führt der Verzicht auf die syntaktische Verallgemeinerung zu einem größeren Grammatikumfang für die einzelnen Domänen [JM00]. Typischerweise werden die Produktionsregeln semantischer Grammatiken deshalb auch nur auf den benötigten Abstraktionsgrad beschrieben und nicht weiter generalisiert. Die erstellten semantischen Grammatiken sind größtenteils vom modellierten Gegenstandsbereich abhängig und nur schwer auf andere Domänen übertragbar. Es gibt jedoch auch semantische Konzepte, wie Datums- oder Zeitangaben, die relativ universell verwendbar sind. Eine semantische Grammatik könnte

z. B. durch einen Transduktor T_{KG} realisiert werden. Dieser wird aus einer Menge von Konzepttransduktoren T_k konstruiert.

$$T_{KG} = \left(\bigoplus_{k \in K} T_k \right)^* \quad (3.1)$$

Ein einzelner Konzepttransduktor generiert dabei jeweils ein Merkmal bzw. ein *Attribut-Wert-Paar*. Der *Kleenesche Abschluss* ermöglicht eine beliebige Reihenfolge artikulierter semantischer Primitiva. Durch die Verwendung eines „Füllwort“-Modells kann der Benutzer beliebige sprachliche Ausdrücke produzieren.

Problematisch bei diesen Ansätzen ist allerdings die prinzipielle Möglichkeit der Zuordnung von verschiedenen semantischen *Konzepten* zu einer Phrase. Es entstehen somit mehrere Hypothesen für Bedeutungsstrukturen. In diesem Fall sind für eine Entscheidungsfindung zusätzliches Wissen oder Rückfragen an den Benutzer nötig. Denkbar wären die Verwendung dialogschrittabhängiger Grammatiken oder die Nutzung eines Wahrscheinlichkeitsmaßes, welches auf dem gemeinsamen Auftreten von *Konzepten* in Sprachäußerungen der Lernstichprobe basiert. Zudem ist eine Bestimmung der Rangordnung durch die Verwendung vieler unterschiedlicher *semantischer Rahmen* bzw. generierter Schablonen für ungeordnete Mengen von *Attribut-Wert-Paaren* möglich. Dieser Ansatz wird z. B. auch im *CMU Phoenix Parser* [War91, WI94] verwendet. Die freien Positionen der *Frames* werden zunächst mit den detektierten semantischen Einheiten gefüllt. Anschließend wird der *semantische Rahmen* ausgewählt, der zu der höchsten Wortanzahl der Sprachäußerung zugeordnet werden kann.

Auch die in dieser Arbeit verwendeten ungewichteten *Äußerungs-Bedeutungs-Transduktoren* sind noch in die Kategorie der semantischen Grammatiken einzuordnen. Die auf Grundlage des bedeutungsorientierten Sprachmodells erstellten stochastischen Grammatiken gehören hingegen in die Klasse der datenbasierten, statistischen Verfahren. Ausführlicher werden die *Äußerungs-Bedeutungs-Transduktoren* im Kapitel 4.3 beschrieben. Problematisch bei *semantischen Grammatiken* für längere Wortfolgemuster (komplette Satz- bzw. Äußerungsstruktur) ist der große Umfang zu berücksichtigender Sprachzeichenfolgen. Hierbei werden Verfahren benötigt, welche den Modellierungsaufwand reduzieren.

3.1.3 Statistikbasierte Interpretationsverfahren

Korpusstatistische Verfahren zur semantischen Interpretation lösen das Problem der Akquise von linguistischem Fachwissen, welches für die Grammatikerstellung benötigt wird. Das sprachliche Wissen wird in diesen Fällen durch die verwendeten Daten indirekt modelliert. Ungenauigkeiten der gesprochenen Sprache können dabei mit Hilfe von Wahrscheinlichkeitsverteilungen beschrieben werden. Somit ist es heute möglich,

innerhalb kurzer Zeit recht robuste Systeme zu entwickeln. Voraussetzung dafür ist eine geeignete und konsistent erstellte Lernstichprobe. In den meisten Fällen werden semantisch beschriftete Sprachdaten benötigt. Aber auch vorgegebene Paare von Sprachäußerung und dazugehöriger Semantikrepräsentation werden zum Training semantischer Modelle genutzt.

Viele aktuelle statistikbasierte Verfahren führen zur Interpretation zwei aufeinanderfolgende Teilschritte durch [Din10]:

1. Zuordnung einer semantischen Konzeptbeschriftung $\hat{\mathbf{t}}$ zu der Wortfolge \mathbf{w} einer sprachlichen Äußerung
2. *Attribut-Wert-Extraktion* aus semantisch segmentierter Sprachzeichenfolge $(\mathbf{w}, \hat{\mathbf{t}})$

Konzeptbeschriftung

Der Prozess der Bestimmung einer Konzeptmarkierungsfolge zu einer Sprachzeichenfolge wird hier als *Konzeptbeschriftung* bezeichnet. Für diese Aufgabe können bereits bekannte Verfahren zur Etikettierung von sequenziellen Strukturen verwendet werden. Durch die Klassifikation erfolgt eine direkte Zuordnung einer semantischen Beschriftung zu einem Sprachzeichen. Da ein *Konzept* durch mehrere Wörter repräsentiert werden

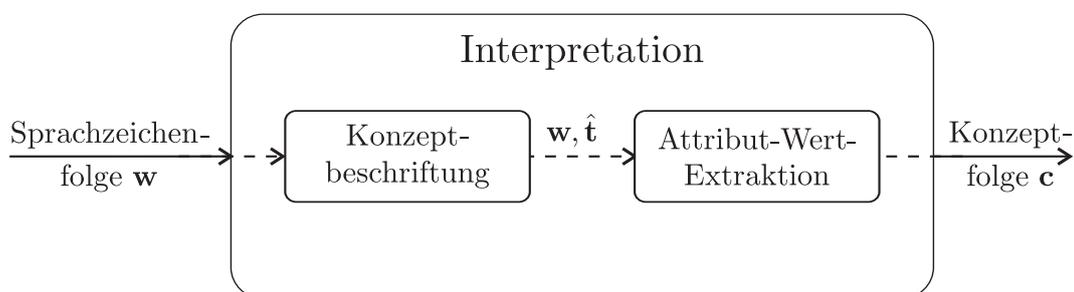


Abb. 3.4: Zweistufiger Interpretationsprozess - Attribut-Wert-Extraktion zu der klassifizierten Konzeptmarkierungsfolge $\hat{\mathbf{t}}$

kann, werden hierbei Konzeptmarkierungen verwendet. Ein Klassifikator ordnet jedem beobachteten Wort eine dazugehörige Konzeptbeschriftung zu. Durch die ermittelten Konzeptgrenzen ist eine anschließende semantisch-orientierte Segmentierung von Wortfolgen möglich. Gewöhnlich erfolgt eine Beschriftung mit Hilfe von *IOB*-Markierungen (*Inside*, *Outside*, *Beginning*). *IOB*-Tags wurden in [RM95] vorgeschlagen und werden vor allem auf dem Gebiet der Informationsextraktion verwendet. Mittlerweile gibt es allerdings verschiedene Versionen und Erweiterungen, so dass in der Literatur keine einheitlichen Darstellungen zu finden sind. Die Markierung des Anfangs einer detektierten Folge (Chunk) erfolgt durch die Kennzeichnung *B*. In einem Block aus mehreren Sprachzeichen werden die nachfolgenden Zuordnungen mit einem *I* markiert. Wenn kein semantisches *Konzept* klassifiziert werden kann, wird dies durch die Outsidemarkierung *O* notiert. In [HDR⁺11] werden diese z. B. durch die einheitliche Attributbezeichnung

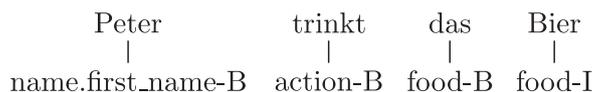


Abb. 3.5: Konzeptmarkierungssequenz zu der Wortfolge „Peter trinkt das Bier“

„null“ ersetzt. Eine Folge von Konzeptbezeichnungen mit *IOB*-Tags wird nachfolgend als *Konzeptmarkierungssequenz* \mathbf{t} bezeichnet. Abbildung 3.5 zeigt eine mögliche Zuordnung einer *Konzeptmarkierungssequenz* zu einer gegebenen Sprachzeichenfolge. Die auf diesem Weg semantisch segmentierten Phrasen dienen zu Bestimmung der normalisierten Datenwerte.

Attribut- Wert-Extraktion

In einem Sprachdialogsystem ist neben der Bestimmung der Merkmalbezeichnung auch ein Normalisierung der Werte notwendig. Dabei repräsentiert der normalisierte Datenwert eine Menge von Wörtern oder Phrasen der analysierten Sprache. Durch die einheitliche Notation von Attribut- und Wertbezeichnung sind beispielsweise Datenbankabfragen möglich. Die Attributbezeichnung kann in der Regel direkt aus der Konzeptmarkierungsfolge \mathbf{t} abgeleitet werden. Für eine Bestimmung der dazugehörigen Datenwerte dient die zu dem Attribut segmentierte Phrase der Wortfolge \mathbf{w} . Hierbei müssen u. a. unterschiedliche sprachlichen Varianten klassifiziert und numerische Angaben in einem einheitlichen Format repräsentiert werden. Einige Wörter, wie z. B. Eigennamen können direkt übernommen werden. Die *Merkmal-Wert-Extraktion* erfolgt üblicherweise regelbasiert [HDR⁺11]. Hierfür können z. B. endliche Transduktoren oder entsprechende Skripte verwendet werden.

Typische Interpretationsverfahren

Im Folgenden werden einige diskriminative und generative Modelle zur Ermittlung semantischer *Konzepte* kurz vorgestellt. Einen Schwerpunkt stellt die Interpretation mit endlichen Automaten dar, da dieser Aspekt für den späteren Verlauf der Arbeit von besonderer Bedeutung ist. Viele der hier aufgeführten statistischen Modelle, wie z. B. *Conditional Random Fields*, *Maximum-Entropie Markovmodelle*, *maschinelle Übersetzungsverfahren* oder *neuronale Netze* werden auch in anderen Bereichen der Sprachverarbeitung eingesetzt.

Conditional Random Field

Conditional Random Fields (CRF) [LMP01] sind graphische Modelle, welche zur Beschriftung oder zum Parsen von Zeichenketten verwendet werden. Zum ersten Mal wurden sie in [RR07] zur semantischen Etikettierung von Wortfolgen (engl. *concept labelling*) genutzt. Dieses diskriminative merkmalsbasierte Verfahren faktorisiert die bedingte Wahrscheinlichkeit $P(\mathbf{t}|\mathbf{w})$. Zu der gegebenen Sprachzeichenfolge \mathbf{w} der Länge N und der zu bestimmenden Konzeptmarkierungssequenz $\mathbf{t} = t_1, \dots, t_N$ ergibt sich

die Verteilung $P(\mathbf{t}|\mathbf{w})$ für ein CRF mit linearer Struktur (*Linear-Chain Conditional Random Field*) durch

$$P(\mathbf{t}|\mathbf{w}) = \frac{1}{Z} \prod_{n=1}^N \exp \left(\sum_{k=1}^K \lambda_k \cdot m_k(t_{n-1}, t_n, \mathbf{w}) \right). \quad (3.2)$$

Zur Bestimmung eines Wahrscheinlichkeitsmaßes wird der wortfolgenabhängige Normierungsterm Z verwendet. Dieser berücksichtigt alle möglichen Konzeptmarkierungssequenzen $\tilde{\mathbf{t}}$ der Länge N zu der gegebenen Wortfolge \mathbf{w} und kann durch

$$Z = \sum_{\tilde{\mathbf{t}}} \prod_{n=1}^N \exp \left(\sum_{k=1}^K \lambda_k \cdot m_k(\tilde{t}_{n-1}, \tilde{t}_n, \mathbf{w}) \right) \quad (3.3)$$

bestimmt werden [HDR⁺11]. Typischerweise wird eine große Anzahl binärer Indikatorfunktionen $m_k(t_{n-1}, t_n, \mathbf{w}) \in \{0, 1\}$ (Merkmal detektiert oder nicht) verwendet. Diese Eigenschaften müssen automatisch aus der Lernstichprobe ermittelt werden. Dabei kommen recht unterschiedliche Merkmalmuster (siehe z. B. [BMON03, Din10]) zum Einsatz, welche Abhängigkeiten zwischen aktueller Konzeptbeschriftung und der kompletten Beobachtungsfolge ($m_k(t_n, \mathbf{w})$) sowie zu den Zustandsübergängen ($m_k(t_{n-1}, t_n)$) beschreiben. Eine mögliche Merkmalfunktion könnte beispielsweise

$$m(t_n, w_{n-1}) = \begin{cases} 1 & \text{wenn } t_n = \text{name.first_name-B} \wedge w_{n-1} = \text{der} \\ 0 & \text{sonst} \end{cases} \quad (3.4)$$

sein. Ein *Conditional Random Field* kann als Faktorgraph [KFL01] über \mathbf{w} und \mathbf{t} dargestellt werden. Dies ist ein bipartiter, ungerichteter Graph $G = (V, F, E)$ mit einer Menge von Faktorknoten F , einer Kantenmenge E sowie der Knotenmenge $V = W \cup T$ zu möglichen Wörtern und Konzeptbeschriftungen. Die Faktoren ergeben sich aus den lokalen Potentialfunktionen f_n . Diese sind nach Gleichung 3.2 durch

$$f_n(t_{n-1}, t_n, \mathbf{w}) = \exp \left(\sum_{k=1}^K \lambda_k \cdot m_k(t_{n-1}, t_n, \mathbf{w}) \right) \quad (3.5)$$

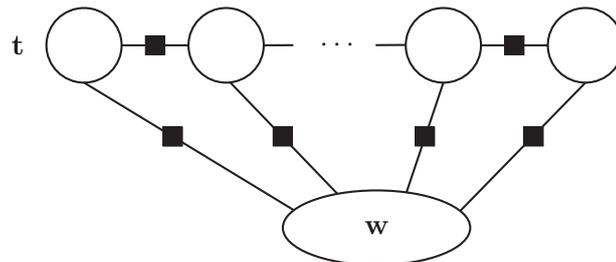


Abb. 3.6: Vereinfachtes graphisches Modell eines Linear-Chain Conditional Random Fields nach [SM12]. Faktorknoten sind hier als ausgefüllte Quadrate dargestellt.

gegeben. Da Faktoren jeweils Informationen der gesamten Wortfolge \mathbf{w} nutzen können, sind die Knoten der beobachteten Zustandsfolge in der Abb. 3.6 vereinfacht als ein gemeinsamer Knoten zusammengefasst. Der Parametervektor $\vec{\lambda} \in \mathbb{R}^K$ dient zur Gewichtung der Menge von Merkmalen. Anhand der Trainingsdaten werden mit dem Maximum-Likelihood-Ansatz die optimalen Parameter bestimmt [SM12]. Die Entscheidung für eine Konzeptmarkierungssequenz zu der gegebenen Sprachzeichenfolge \mathbf{w} erfolgt letztendlich durch

$$\hat{\mathbf{t}} = \arg \max_{\mathbf{t}} \{P(\mathbf{t}|\mathbf{w})\}. \quad (3.6)$$

Hierbei kann die Normalisierungskonstante Z vernachlässigt werden [HDR⁺11].

Eine Möglichkeit der gemeinsamen Dekodierung von Dialogakttyp und Konzeptmarkierungsfolge zu einer gegebenen Sprachzeichensequenz wird in [JL08] vorgestellt. Die dazu neu entwickelte Modellstruktur ist als *triangular-chain CRF* bezeichnet worden.

Maximum-Entropie Markovmodell

Wie das *Conditional Random Field* gehört auch das *Maximum-Entropie Markovmodell (MEMM)* [MFP00, BMON03] zu den log-linearen Modellen. Die Zustandsübergangswahrscheinlichkeiten eines *Maximum-Entropie Markovmodells* werden in Abhängigkeit von den Wahrscheinlichkeiten des aktuellen Zustands und der Beobachtungsfolge \mathbf{w} bestimmt. Für dieses gerichtete graphische Modell (vgl. Abb. 3.7) können dieselben Merkmalfunktionen $m_k(t_n, \mathbf{w})$ eines CRF verwendet werden. Der wesentliche Unterschied zu

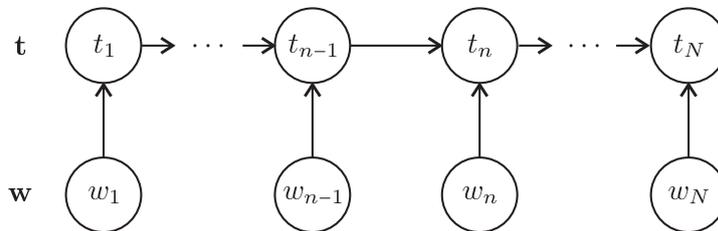


Abb. 3.7: Graphische Struktur eines Maximum-Entropie Markovmodells nach [LMP01]

den *Conditional Random Fields* besteht darin, dass für jeden Zustand der Konzeptmarkierungssequenz ein exponentielles Modell benötigt wird [LMP01]. Somit unterscheidet sich hier auch der Normierungswert Z zur Bestimmung des Wahrscheinlichkeitsmaßes. Unter Verwendung der Formel 3.2 kann dieser durch

$$Z = \prod_{n=1}^N \sum_{\tilde{t}} \exp \left(\sum_{k=1}^K \lambda_k \cdot m_k(t_{n-1}, \tilde{t}_n, \mathbf{w}) \right) \quad (3.7)$$

berechnet werden. Im Vergleich zu Formel 3.3 wird die Summe über alle \tilde{t} , anstatt über alle möglichen Sequenzen $\tilde{\mathbf{t}}$ gebildet. Als Entscheidungskriterium wird ebenfalls Gleichung 3.6 verwendet [HDR⁺11].

Durch die Verteilung der Wahrscheinlichkeitsmasse auf die ausgehenden Zustandsübergänge werden tendenziell die Zustände bevorzugt, welche eine geringere Anzahl an

Transitionen besitzen [LMP01]. In praktischen Untersuchungen, wie z. B. in [HDR⁺11], werden mit *Maximum-Entropie Markovmodellen* im Vergleich zu den CRFs schlechtere Klassifikationsergebnisse erzielt.

Supportvektormaschinen

Durch eine Folge von Klassifikationsvorgängen kann auch mittels Supportvektormaschinen (SVM) [CV95] die semantische Beschriftung einer gegebenen Sprachzeichensequenz erreicht werden. Für die Untersuchungen in [RR07] wurde die Open-Source-Software *YamCha* [KM01] verwendet. Zu jeder Konzeptbeschriftungsklasse gibt es in den Trainingsdaten eine Menge von Repräsentanten. Hierbei wird für jedes zu Paar (t_i, w_i) ein Merkmalvektor berechnet. Die Komponenten des Vektors werden z. B. aus den vorherigen und nachfolgenden Wörtern (typisch: w_{i-2}^{i+2}) oder benachbarter Konzeptbeschriftungen abgeleitet. *Supportvektormaschinen* sind binäre Klassifikatoren. Für eine Beschriftung eines Wortes werden hier paarweise Entscheidungen durchgeführt. Dazu müssen für K mögliche Klassen $0,5 \cdot K(K - 1)$ Klassifikatoren separat trainiert werden. Als Ergebnis wird sich für diejenige Konzeptmarkierung zu einem Wort entschieden, welche am häufigsten klassifiziert wurde [KM01].

Semantic Tuple Classifier

In [MGJ⁺09] wird ein diskriminatives Klassifikationsverfahren vorgestellt, welches zu der analysierten Wortfolge eine Menge von Tupel mit semantischen Beschriftungen liefert. Aus diesen Tupeln wird in einem Nachbearbeitungsschritt eine semantische Baumstruktur erzeugt.

Als Trainingsdaten wird eine Menge von Paaren, jeweils bestehend aus einer Sprachäußerung und einem dazugehörigen semantischen Baum, benötigt. Jeder dieser Bäume enthält dabei nur die Merkmale und Datenwerte, welche in der verwendeten Datenbank berücksichtigt werden können. Einige Attribute sind mit sehr vielen Datenwerten assoziiert. Bei der Erstellung der Lernstichprobe und für den Klassifikationsprozess werden deshalb ausgewählte Wörter durch ihre Kategoriebezeichnung ersetzt. Sollten Kategoriebezeichnungen mehrfach in einer Sprachäußerung auftreten, so werden diese der Reihenfolge nach mit einem Index versehen.

Für das Training der *Supportvektormaschinen* ist einer Zerlegung des semantischen Baumes in Tupel erforderlich. Für jedes individuelle Tupel wird ein binärer Klassifikator erzeugt. Eine Sprachäußerung der Lernstichprobe gehört zu der individuellen Klasse, wenn das semantische Tupel mit ihr assoziiert ist. Als Merkmale der Sprachäußerung werden N-Gramm-Häufigkeiten verwendet. Zusätzlich wird ein Multi-Klassen SVM-Modell für die Bestimmung des Dialogakttyps eingesetzt. Die dekodierte Beschriftung bildet die Wurzel des semantischen Baums. Die zu der Sprachäußerung klassifizierte Menge semantischer Tupel wird für die Generierung der Bedeutungsstruktur verwendet (vgl. Abbildung 3.8) Das erste Wort jedes Tupels bestimmt den dazugehörigen Elternknoten. Tupel, die sich auf diese Weise nicht der Bedeutungsstruktur zuordnen lassen, werden

Sprachäußerung: $FIRST_NAME(Peter)$ trinkt das Bier

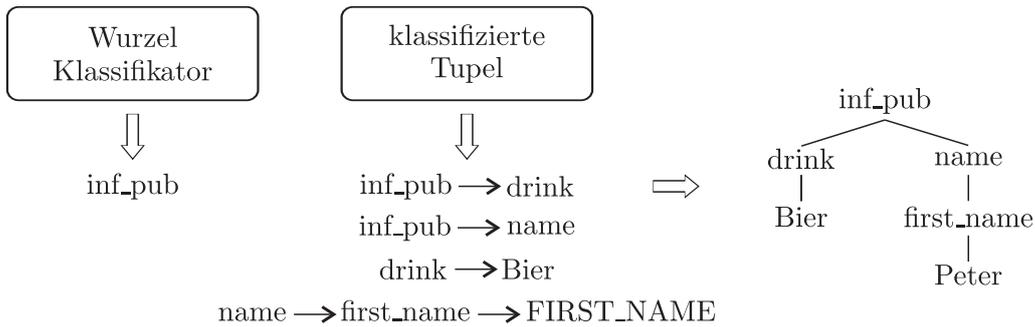


Abb. 3.8: Ableitung eines semantischen Baums zu der positiv klassifizierten Tupelmengung. In der zu klassifizierenden Wortfolge wird hier der Vorname Peter durch die Kategorie $FIRST_NAME$ substituiert.

mit einer domänenabhängigen Grammatik verarbeitet. Abschließend werden Blätter des Baumes durch Datenwerte (z. B. $FIRST_NAME$ durch $Peter$) ersetzt [MGJ⁺09].

Rekurrente neuronale Netze

Auch *rekurrente neuronale Netze* können für die Bestimmung von Konzeptmarkierungsfolgen zu einer gegebenen Sprachzeichenfolge verwendet werden. Untersuchungen zur Verwendung verschiedener Architekturen und Konfigurationen von neuronalen Netzen findet man beispielsweise in [MHDB13, YZH⁺13, MDY⁺15]. Anhand des Elman-Netzes [Elm90] soll hier nur das Grundprinzip der Verwendung neuronaler Netze zur Bestimmung der wahrscheinlichsten Sequenz \hat{t} beschrieben werden. Eine schematische Darstellung der Architektur eines solchen Elman-Netzwerks ist durch Abb. 3.9 gegeben. Zum Zeitpunkt k erhält die verborgene Schicht zusätzlich die Ausgabe des eigenen Inhalts vom vorherigen Zeitpunkt ($k - 1$). Dieser rückgekoppelte Inhalt erhält somit auch Informationen zur Historie.

Die gelernten Netzgewichte sind durch die Matrizen \mathbf{U} , \mathbf{V} und \mathbf{W} gegeben. Das zum

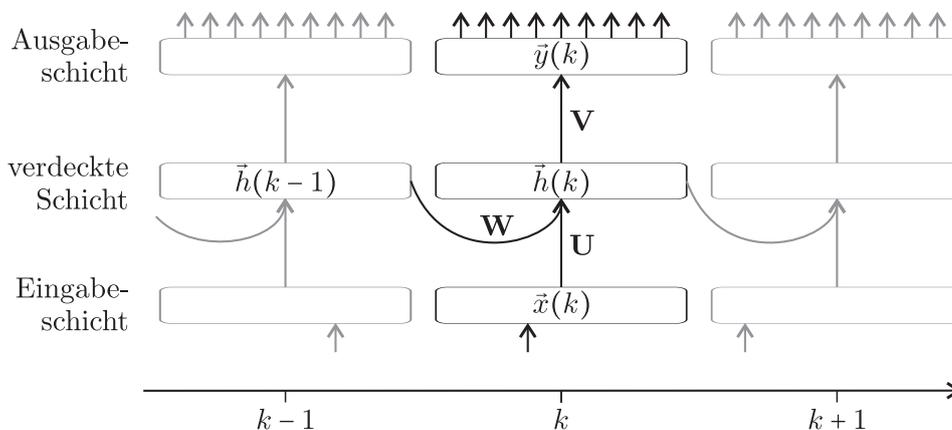


Abb. 3.9: Zeitlich „ausgerolltes“ rekurrentes neuronales Netz (Elman-Architektur) für drei aufeinanderfolgende Eingabewörter nach [Elm90, YZH⁺13]

Zeitpunkt k vorliegenden Wort der Sprachzeichenfolge wird durch die Eingabeschicht repräsentiert. Diese ist durch den Vektor $\vec{x}(k)$ beschrieben, wobei die Dimension des Vektors der Vokabulargröße entspricht. Im Vektor $\vec{x}(k)$ besitzt nur die Komponente den Wert 1, dessen Index dem zu analysierenden Wort zugeordnet ist. Die anderen Werte sind entsprechend 0 [YZH⁺13].

Die Ausgabeschicht eines solchen neuronalen Netzes liefert zum diskreten Zeitpunkt k den Vektor $\vec{y}(k)$. Die Anzahl L der Vektorkomponenten entspricht der Anzahl möglicher Konzeptbeschriftungen (Klassen). Der nicht von außen beobachtbare Vektor $\vec{h}(k)$ sowie der Ausgabevektor $\vec{y}(k)$ ergeben sich durch

$$\vec{h}(k) = f(\mathbf{U} \cdot \vec{x}(k) + \mathbf{W} \cdot \vec{h}(k-1)) \quad (3.8)$$

$$\vec{y}(k) = g(\mathbf{V} \cdot \vec{h}(k)). \quad (3.9)$$

In [YZH⁺13] wird zum Training des neuronalen Netzes der Backpropagation-Algorithmus verwendet. Als Transferfunktion für die Menge von Neuronen der verdeckten Schicht wird die Sigmoidfunktion

$$f(z) = \frac{1}{1 + e^{-z}} \quad (3.10)$$

genutzt. Für die L Ausgabeneuronen kommt hingegen die sogenannte Softmax-Funktion

$$g(z_m) = \frac{e^{z_m}}{\sum_{i=1}^L e^{z_i}} \quad (3.11)$$

zum Einsatz, wobei sich z_m auf das m -te Element der Ausgabeschicht bezieht. Dadurch wird die Summe der Werte von $\vec{y}(k)$ auf eins normiert und $\vec{y}(k)$ beschreibt eine Wahrscheinlichkeitsverteilung zu der Menge von Konzeptbeschriftungen. Eine Maximumsuche im Ausgabevektor $\vec{y}(k)$ liefert direkt das Klassifikationsergebnis $t(k)$ zu dem analysierten Eingabewort [YZH⁺13, MDY⁺15].

Eine Erweiterungsmöglichkeit zu dem beschriebenen Ansatz ist die Berücksichtigung der direkten Vorgänger und Nachfolger des zu klassifizierenden Wortes [MDY⁺15] oder nur der nachfolgenden Sprachzeichen [YZH⁺13] für die Eingabeschicht. Aktuelle Verfahren nutzen zudem Vektorrepräsentationen für die zu analysierenden Wörter. Ein Vektor repräsentiert das jeweilige Sprachzeichen in einem Vektorraum. Wörter, die syntaktisch oder semantisch verwandt sind, besitzen dabei auch eine Nähe im Repräsentationsraum [VRG15]. Die Bestimmung der reellwertigen Vektoren kann zuvor anhand großer Datenmengen, wie z. B. Wikipedia, separat erfolgen [MDY⁺15]. Eine vergleichende Untersuchung aktueller Verfahren zur Generierung von wortrepräsentierenden Vektoren erfolgt z. B. in [MCCD13].

Zusätzliche Merkmalvektoren werden in [YZH⁺13] verwendet. Diese sind über separate Gewichtungsmatrizen in die verdeckte Schicht und die Ausgabeschicht eingekoppelt. In Abhängigkeit vom Eingabewort berücksichtigen die Merkmalvektoren z. B. die Wortklasse oder die syntaktische Kategorie (*Part-of-speech*).

Methoden der maschinellen Übersetzung

Auch Verfahren aus dem Bereich der *statistischen maschinellen Übersetzung* werden zur Bestimmung von Attributbezeichnungen zu einer Sprachzeichenfolge verwendet. Die z. B. in [EPR⁺96, PERW97] verwendeten Modelle zur Übersetzung des Quellsatzes in eine formale Zeichenfolge basieren auf dem klassischen generativen Ansatz [BPPM93]. Die Dekodierung der wahrscheinlichsten Sequenz $\hat{\mathbf{t}}$ erfolgt auf diesen Weg durch

$$\hat{\mathbf{t}} = \arg \max_{\mathbf{t}} (P(\mathbf{t}|\mathbf{w})) = \arg \max_{\mathbf{t}} (P(\mathbf{w}|\mathbf{t}) \cdot P(\mathbf{w})). \quad (3.12)$$

In [PRW97] werden zusätzliche binäre Merkmalwerte zur Bestimmung der A-posteriori Wahrscheinlichkeit verwendet. Die phrasenbasierten Merkmale werden hier anhand der Korrelation der Sprachzeichen von Quell- und formaler Zielsprache bestimmt.

In [HDR⁺11] wird ein statistisches Übersetzungsverfahren verwendet, welches auf dem *Maximum-Entropie-Ansatz* [BPP96] basiert. Abbildung 3.10 zeigt das Grundprinzip dieses Ansatzes. Hierbei werden gleich mehrere Modelle zur Ermittlung der Konzeptmarkierungssequenz genutzt. Jede Modellbewertung erhält zusätzlich einen Gewichtungsparemeter λ_k . Für die Übersetzung kommen u. a. phrasenbasierte Übersetzungsmodelle für beide Übersetzungsrichtungen und ein Sprachmodell der Konzeptmarkierungssprache zum Einsatz. Die für die Lernstichprobe optimalen Parameter $\{\lambda_k\}$ werden für den Übersetzungsprozess verwendet [HDR⁺11]. Nach [ON02] ergibt sich die A-posteriori

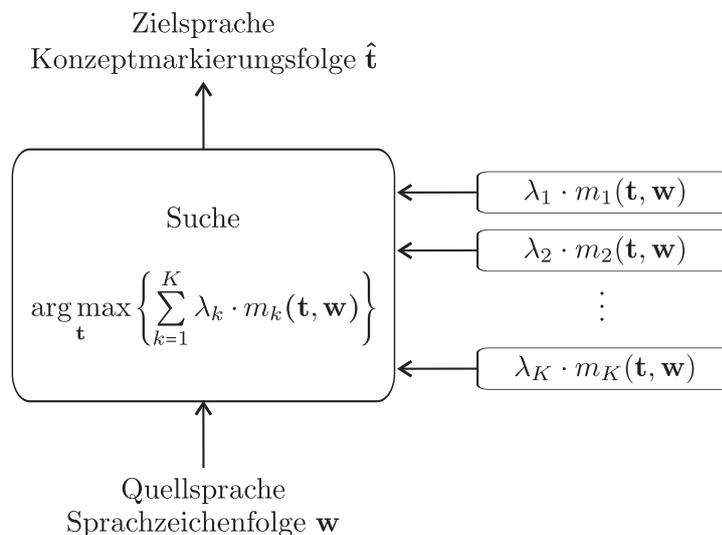


Abb. 3.10: Prinzip eines log-linearen Übersetzungsmodells nach [Mac09]

Wahrscheinlichkeit mit

$$P(\mathbf{t}|\mathbf{w}) = \frac{\exp\left(\sum_{k=1}^K \lambda_k \cdot m_k(\mathbf{t}, \mathbf{w})\right)}{\sum_{\tilde{\mathbf{t}}} \exp\left(\sum_{k=1}^K \lambda_k \cdot m_k(\tilde{\mathbf{t}}, \mathbf{w})\right)}. \quad (3.13)$$

Durch die Verwendung der Exponentialfunktion in Gleichung 3.13 können die gewichteten logarithmierten Bewertungen der Teilmodelle addiert werden. Zur Berechnung eines Wahrscheinlichkeitswertes dient der Normierungsfaktor Z . Dieser entspricht dem Nenner in Formel 3.13 und berücksichtigt alle realisierbaren Konzeptmarkierungsfolgen $\tilde{\mathbf{t}}$ zur Sprachzeichenfolge \mathbf{w} . Eine Bestimmung der optimalen Hypothese $\hat{\mathbf{t}}$ erfolgt durch

$$\hat{\mathbf{t}} = \arg \max_{\mathbf{t}} \{P(\mathbf{t}|\mathbf{w})\} = \arg \max_{\mathbf{t}} \left\{ \sum_{k=1}^K \lambda_k \cdot m_k(\mathbf{t}, \mathbf{w}) \right\}, \quad (3.14)$$

Die monoton steigende Exponentialfunktion sowie der konstante Normierungsfaktor haben keinen Einfluss auf die zu treffende Entscheidung.

Hidden Vector State Modell

Dieses Verfahren basiert auf einer probabilistischen kontextfreien Grammatik und wird in [HY05] beschrieben. Basis dafür ist die Verwendung von semantischen Vektoren, welche die übergeordneten semantischen Kategorien definieren. Bei der Erstellung der Lernstichprobe wird jeder Sprachäußerung eine Semantikstruktur zugeordnet. Zusätzlich werden einige Wörter bzw. Phrasen durch domänenspezifische Wortklassen

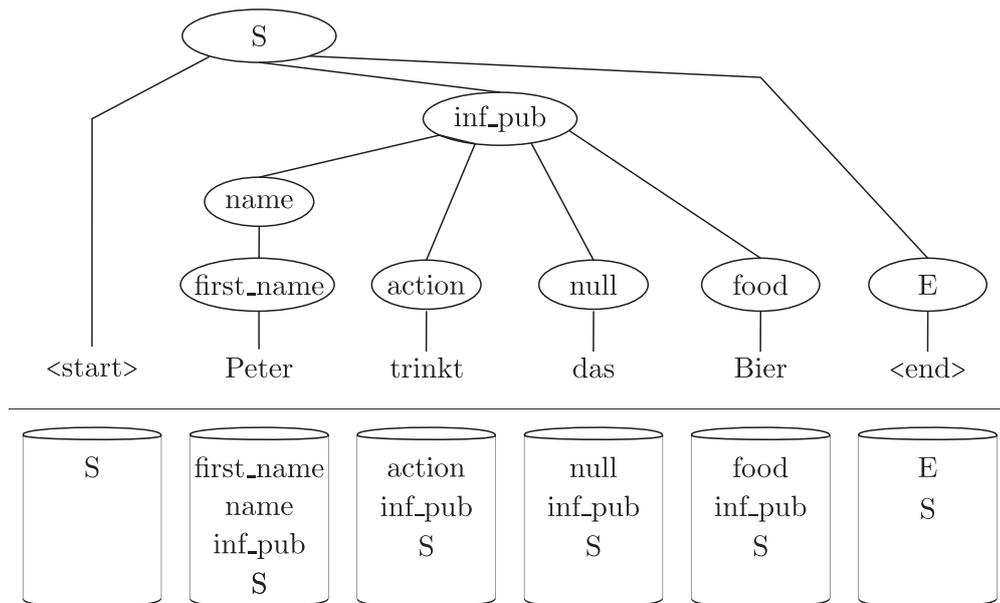


Abb. 3.11: Vergleich eines möglichen semantischen Ableitungsgraphen (oben) zu der Wortfolge „Peter trinkt das Bier“ und der dazugehörigen Sequenz von semantischen Vektoren (unten) nach [HY05]

(z. B. „Peter“ = `first_name`) ersetzt.

Bei der Klassifikation erhält man eine Vektorfolge \vec{v} . So wird beispielsweise dem Zeichen „Peter“ der Zustandsvektor $\vec{v}(\text{Peter}) = [\text{first_name}, \text{name}, \text{inf_pub}, \text{S}]$ zugeordnet. Die zu einer Sprachäußerung ermittelte Vektorfolge \vec{v} kann durch eine äquivalente Baumstruktur dargestellt werden. Dieser Zusammenhang ist in Abb. 3.11 dargestellt.

Der in diesem Ansatz verwendete semantische Parser ist mit einem Kellerautomaten vergleichbar, bei dem der Inhalt des Stapelspeichers einem semantischen Vektor entspricht. Jeder semantische Vektor wird dabei durch einen Zustand repräsentiert. Ein Zustandsübergang in dem *Hidden Vector State Modell* ist durch eine Menge von Verarbeitungsschritten („push“ und „pop“ von Merkmalbezeichnungen) und der Generierung eines neuen Wortes gekennzeichnet [HY05]. Als Ergebnis dieses semantischen Parsingprozesses erhält man eine Merkmalstruktur. Eine Normalisierung von Datenwerten wird auf diesem Weg nicht erreicht.

Dynamische Bayes'sche Netze

Für die semantische Interpretation werden in [Lef06] *Dynamische Bayes'sche Netze* (DBN) eingesetzt. Dieses generative Verfahren ermöglicht eine direkte Bestimmung der wahrscheinlichsten Konzeptfolge \hat{c} (= gemeinsame Attribut-Wertsequenz $(\hat{\mathbf{a}}, \hat{\mathbf{v}})$) zu der beobachteten Wortfolge \mathbf{w} . Eine Entscheidung ergibt sich hierbei durch

$$\begin{aligned} \hat{\mathbf{a}}, \hat{\mathbf{v}} &= \arg \max_{\mathbf{a}, \mathbf{v}} P(\mathbf{a}, \mathbf{v} | \mathbf{w}) \\ &= \arg \max_{\mathbf{a}, \mathbf{v}} P(\mathbf{w} | \mathbf{a}, \mathbf{v}) P(\mathbf{v} | \mathbf{a}) P(\mathbf{a}). \end{aligned} \quad (3.15)$$

Für die Untersuchungen in [Lef06] wurde der semantisch annotierte MEDIA-Korpus genutzt. In diesem sind für die Phrasen der Sprachäußerungen die Attributbezeichnung und der dazugehörige normalisierte Datenwert zugeordnet (vgl. [BMQD09]). In Abhängigkeit von der Wortanzahl einer Sprachzeichenfolge erfolgt in einem *Dynamischen Bayes'schen Netz* die Wiederholung bzw. das sogenannte „Ausrollen“ des statischen Modells. Für die Berechnung der Übergangswahrscheinlichkeiten zwischen den Zufallsvariablen in dem DBN werden bedingte Wahrscheinlichkeitstabellen verwendet. Diese Tabellen werden direkt anhand der Auftretenshäufigkeit in den Trainingsdaten erstellt.

Praktisch erfolgt hier keine gemeinsame Bestimmung der Attribut-Wertsequenz. Zunächst wird zu der beobachteten Wortfolge \mathbf{w} die wahrscheinlichste Attributbezeichnungsfolge \mathbf{a} gesucht. Unter der Annahme dass die Datenwerte nur einen sehr kleinen oder gar keinen Einfluss auf diese Entscheidung haben, ergibt sich die optimale Attributfolge mit Formel 3.15 und der Marginalisierung von \mathbf{v} durch

$$\hat{\mathbf{a}} = \arg \max_{\mathbf{a}} \sum_{\mathbf{v}} P(\mathbf{w} | \mathbf{a}, \mathbf{v}) P(\mathbf{v} | \mathbf{a}) P(\mathbf{a}). \quad (3.16)$$

Teilweise können mehrere aufeinanderfolgende Wörter einem Merkmal zugeordnet

werden. Die Folge von Attributbezeichnungen wird in diesem Modell deshalb auch durch eine sogenannte „Transitionsvariable“ beeinflusst. Ist der Wert dieser Variable gleich eins, so erfolgt die Ermittlung einer neuen Merkmalbezeichnung mit Hilfe der bedingten Wahrscheinlichkeit $P(a_n|a_{n-1})$. Ist der Wert dieses Vorgängerknosens hingegen gleich null, so wird das Attribut für den nächsten Zeitabschnitt wiederholt.

In dem zweiten Schritt erfolgt die Ermittlung der normalisierten Datenwerte anhand der zuvor ermittelten Zustandsfolgen zu Wort-, Attribut- und Transitionssequenz. Die optimale Datenwertsequenz zu der dekodierten Attributfolge $\hat{\mathbf{a}}$ wird durch

$$\hat{\mathbf{v}} = \arg \max_{\mathbf{v}} P(\mathbf{w}|\hat{\mathbf{a}}, \mathbf{v})P(\mathbf{v}|\hat{\mathbf{a}})P(\hat{\mathbf{a}}) \quad (3.17)$$

bestimmt [Lef06, Din10].

Interpretation mit endlichen gewichteten Transduktoren

Die Zuordnung einer Bedeutungsrepräsentation zu einer Wortfolge erfolgt mit endlichen Transduktoren auf Basis stochastischer Sprachmodelle. Endlichen Automaten werden nicht nur für diesen Übersetzungsprozess genutzt, sondern u. a. auch zur Spracherkennung. Eine einheitliche Verwendung dieser Technologie bietet Vorteile gegenüber anderen aufgeführten Ansätzen in diesem Kapitel. Auf diese Weise kann eine parallele Verarbeitung und Bewertung mehrere Wortfolgehypthesen erfolgen [RBMD06]. Zudem stellt die Komposition kompatibler Transduktoren (siehe Kapitel 2.4.2) eine effektive Schnittstelle zwischen verschiedenen Funktionsblöcken eines technischen Systems dar. Somit wird auch eine spätere Klassifikationsentscheidung ermöglicht.

- *Verfahren mit wortweiser Konzeptbeschriftung*

In diesem Abschnitt werden zwei generative Ansätze vorgestellt, welche zunächst jedem Wort eine Attributbeschriftung zuordnen. Das erste Interpretationsverfahren [PMS⁺01] mit endlichen Automaten wird für ein Sprachdialogsystem (Fahrplanauskunft für Zugverkehr) genutzt. Durch Verwendung eines *Part-of-speech Taggers* werden hier einzelne semantische Einheiten mit Hilfe von Wortartbeschriftungen modelliert. Dabei kommen neben den üblichen lexikalisch-grammatischen Klassen auch zusätzliche semantisch motivierte Kategorien zum Einsatz. Auf diese Weise kann bei einem großen zu berücksichtigenden Vokabular der erforderliche Umfang der Trainingsdaten reduziert werden. Zusätzlich wird anhand des Datenkorpus ein stochastisches Bigramm-Modell für Attributfolgen erstellt. Diese Attributbeschriftungen werden anschließend durch die dazugehörigen endlichen Automaten der einzelnen semantischen Einheiten ersetzt. Dadurch erhält man ein integriertes Modell zur Interpretation [PMS⁺01].

Derselbe Ansatz wird in [SSG⁺02] beschrieben. Allerdings erfolgt hier die Modellierung der regulären stochastischen Modelle zu den semantischen Attributen direkt durch

die zugeordneten Phrasen der Lernstichprobe. Eine Bestimmung von Wortarten findet hierbei also nicht statt.

Durch die *dynamische Programmierung* wird die wahrscheinlichste Sequenz von Attributbezeichnungen zu einer gegebenen Wortfolge bestimmt. Pro Zustandsübergang wird genau eine Attributbezeichnung generiert. Dies wird durch das folgende Beispiel veranschaulicht.

Wortsequenz: *Peter trinkt das Bier*

Attributfolge: *first_name action drink drink*

Mit Hilfe der Attributfolge kann der analysierte Satz semantisch segmentiert werden [SSG+02]. Abschließend erfolgt zu den ermittelten Attribut-Phrasenpaaren eine regelbasierte Generierung eines oder mehrerer Frames [PMS+01].

Ein aktuelles Verfahren zur Ermittlung semantisch beschrifteter Wortfolgen wird in [RR07] kurz beschrieben. Ausgangspunkt ist ein gewichteter *Worthypothesengraph* T_W , welcher durch einen endlichen Automaten repräsentiert wird. In diesem Ansatz werden durch die Komposition einzelner Transduktoren folgende Grundfunktionalitäten nacheinander erreicht:

1. Generierung eines Worthypothesengraphen mit semantischer Beschriftung,
2. Bewertung dieses mit Konzeptmarkierungen versehenen Wortgraphen durch ein stochastisches Sprachmodell T_{KSM} ,
3. Attribut-Wert-Extraktion zur wahrscheinlichsten semantisch annotierten Sequenz.

Durch die Hintereinanderausführung der einzelnen Automaten erhält man als Ergebnis der semantischen Analyse den Transduktor T_{SA} , welcher eine Folge von *Attribut-Wert-Paaren* generiert. Dieser gesamte Interpretationsprozess wird mit der Gleichung

$$T_{\text{SA}} = T_W \circ T_G \circ T_{\text{G2K}} \circ T_{\text{KSM}} \circ T_{\text{MWE}} \quad (3.18)$$

ausgedrückt [Din10, HDR+11]. Der Transduktor T_{KSM} beschreibt ein N -Gramm Modell für semantisch beschriftete Sprachzeichenfolgen. In dieser Arbeit wird dies als stochastisches *konzeptuelles Sprachmodell* (conceptual language model [RR07]) bezeichnet.

Analog zu Formel 2.8 kann die Verbundwahrscheinlichkeit für das gemeinsame Auftreten von Wort- und Konzeptmarkierungsfolge der Länge L durch

$$P(\mathbf{w}, \mathbf{t}) = \prod_{i=1}^L P(w_i, t_i | w_{i-(N-1)}, t_{i-(N-1)} \dots w_{i-1}, t_{i-1}) \quad (3.19)$$

geschätzt werden.

Ein *Generalisierungs-Transduktor* T_G hat in diesem Ansatz die Aufgabe, bestimmte Wörter in Wortkategorien bzw. Wortklassen zu übersetzen. Der wesentliche Grund für

diesen Schritt ist der beschränkte Umfang der Trainingsdaten, welcher in der Regel für die Erstellung des Transduktors T_{KSM} zur Verfügung steht. So sind z. B. für Personennamen, geographische Bezeichnungen oder numerischen Angabe viele unterschiedliche sprachliche Einheiten möglich. Diese sind allerdings nur selten oder gar nicht in der Lernstichprobe enthalten. Folglich müssen die Daten zum Training des *konzeptuellen Sprachmodells* ebenfalls aufbereitet werden. Auch hier sind bestimmte Wörter durch ihre Klassenbezeichnung zu ersetzen. Sprachzeichen, welche keiner Klasse zugehörig sind, müssen dabei natürlich nicht substituiert werden. Der für die Klassifikation verwendete Transduktor T_G gibt diese Wörter wieder unverändert aus [Din10].

Der Transduktor T_{G2K} übersetzt eine Folge von Wörtern / Wortkategorien in Sprachzeichenfolgen, welche mit Konzeptetiketten angereichert sind. Eine Modellierung des Transduktors T_{G2K} erfolgt mit Hilfe von endlichen Automaten zu jedem semantischen *Konzept* c_i . Eine Erstellung dieser Grammatiken kann einerseits manuell erfolgen. Zum Beispiel für Zeit- oder Datumsangaben bietet sich diese Art der Modellierung an. Unter Verwendung der semantisch annotierten Sprachdatensammlung können solche Automaten aber auch automatisch generiert werden. Außerdem kommt auch ein „Füllwort-Modell“ für die Wörter oder Phrasen zum Einsatz, welche für die Anwendung semantisch irrelevant sind [RBMD06, HDR⁺11].

Bei dem Dekodierungsprozess generiert der Automat T_{G2K} semantisch annotierte Sequenzen, welche anschließend durch das stochastische Folgenmodell T_{KSM} bewertet werden. Der Pfad mit den geringsten Kosten in diesem Kompositionsergebnis liefert die wahrscheinlichste Hypothese. Diese Sequenz wird für die *Attribut-Wert-Extraktion* verwendet [Din10]. Dazu dient der Transduktor T_{MWE} , welcher in Abhängigkeit der Konzeptmarkierungsfolge eine Menge von Attributbezeichnungen mit dazugehörigen Datenwerten erzeugt.

Hinweis: Das hier dargestellte Prinzip wurde im Rahmen der Arbeit exemplarisch umgesetzt. Weitere Ausführungen zur Erstellung und zur Verwendung dieser Automaten finden sich im Kapitel 6.1.2. Somit ist eine vergleichende Betrachtung mit den modellierten *Äußerungs-Bedeutungs-Transduktoren* möglich.

- *Graphenbasiertes Verfahren mit phrasenweiser Attributzuordnung*

Ein multilinguales System zur semantischen Interpretation wird in [CHGS12] vorgestellt. Weiterentwicklungen dieses Modells und Untersuchungsergebnisse sind in [CGS13, CHG⁺16] veröffentlicht worden. Für den Informationsaustausch zwischen den Teilmodulen des Gesamtsystems werden gewichtete Graphen verwendet. Die Zustandsübergangsgewichte sind jeweils Wahrscheinlichkeitswerte. Allerdings werden bei diesem graphenbasierten Ansatz zwar Automaten, jedoch keine Transduktoren verwendet. Zudem wird auf den Einsatz typischer Automatenoperationen verzichtet.

Erforderliche semantische Modelle wurden mittels der spanischsprachigen Trainingsdaten des DIHANA-Projekts [BLV⁺06] erstellt. Aus der segmentierten und beschrifteten Lernstichprobe sind Sprachmodelle für jedes Attribut a_n erstellt worden. Diese werden durch endliche Automaten repräsentiert. Stochastische Grammatiken ermöglichen hier die Schätzung der Wahrscheinlichkeit $P(\mathbf{w}_k^l | a_n)$ möglicher Phrasen \mathbf{w}_k^l zu dem jeweiligen Attribut. Weiterhin wird aus der Datensammlung ein Bigramm-Modell für Attributfolgen erstellt, welches zur Bestimmung der wahrscheinlichsten Attribut-Wert-Sequenz (\mathbf{w}, \mathbf{a}) verwendet wird.

In dem multilingualen System wird eine erkannte Sprachäußerung der Quellsprache zunächst durch mehrere maschinelle Übersetzer in Wortfolgen der „Systemsprache“ übersetzt, für die semantische Modelle erstellt wurden. Durch ein multiples Sequenzalignement wird aus den Übersetzungsergebnissen der gewichtete Worthypothesengraph \mathcal{A}_W generiert.

Die semantische Interpretation von \mathcal{A}_W erfolgt durch folgende Schritte:

1. Erstellung eines gewichteten Automaten \mathcal{A}_{WK} mit semantisch beschrifteten Phrasen.
2. Bestimmung und Extraktion der wahrscheinlichsten Beschriftung (\mathbf{w}, \mathbf{a}) . Hierbei erfolgt zusätzlich die Berücksichtigung der Bewertung durch ein Bigramm-Modell für Attributfolgen.
3. Slotzuordnung in einem semantischen Rahmen

Der zu konstruierende Automat \mathcal{A}_{WK} repräsentiert mögliche Sequenzen von Phrasen-Attribut-Paaren (\mathbf{w}_i^j, a_n) zu dem gegebenen Wortgraphen \mathcal{A}_W . Dieser Zusammenhang ist in Abb. 3.12 anhand eines Ausschnitts der Automatengraphen dargestellt. Beide Automaten verwenden dasselbe Zustandsalphabet Z ($Z_W = Z_{WK}$).

Durch den gerichteten azyklischen Graphen \mathcal{A}_W sind für jedes Paar $z_i, z_j \in Z$ mit $z_i < z_j$ eine Menge möglicher zusammenhängender Pfade $\mathcal{U}(z_i, z_j)$ gegeben. Die Bestimmung der Zustandsübergangsmenge E_{WK} erfolgt zu jedem Knotenpaar (z_i, z_j) des

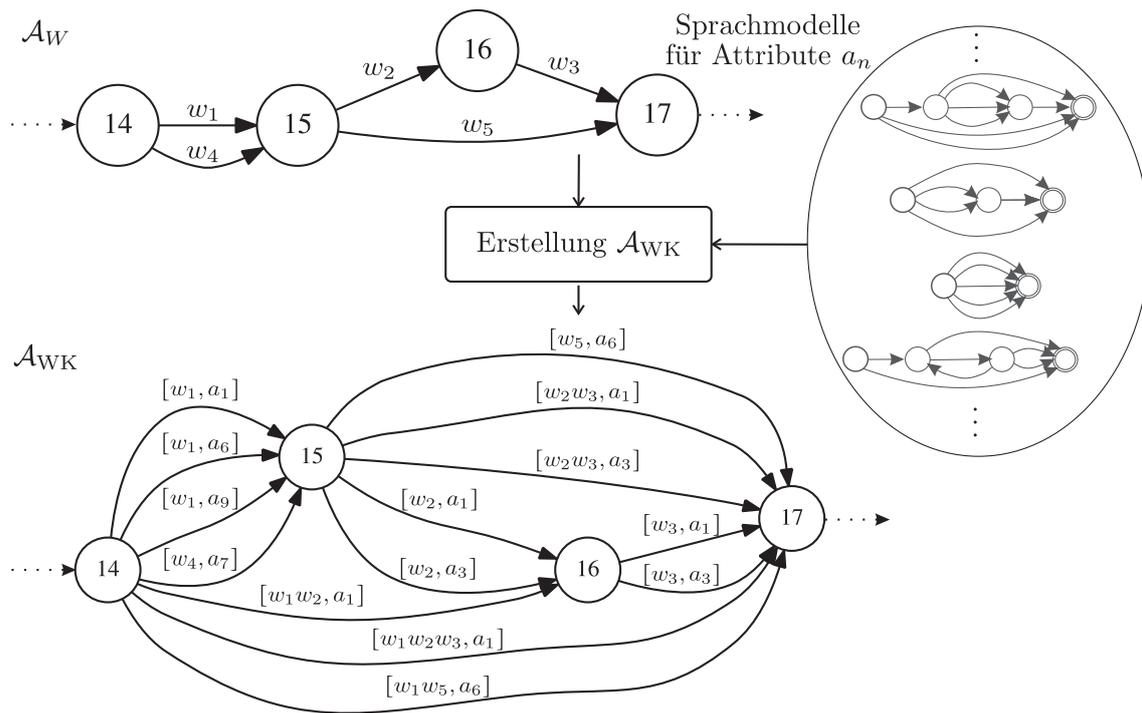


Abb. 3.12: Grundprinzip der Konstruktion des Automatengraphen \mathcal{A}_{WK} zum Worthyphotesgraph \mathcal{A}_W nach [Cal15]. Auf die Darstellung möglicher Wahrscheinlichkeitswerte wird verzichtet.

Automaten \mathcal{A}_{WK} attributabhängig. Zu einer semantischen Kategorie a_n wird anhand der gemeinsamen Bewertung aus den lokalen Abständen im Graphen \mathcal{A}_W und der Wahrscheinlichkeitsbewertung des semantischen Sprachmodells nur die beste Hypothese ausgewählt. Die optimale Phrase gehört zu dem Pfad $\pi(z_i, z_j)$ in \mathcal{A}_W , der das Gesamtgewicht $w_{WK} = w[\pi(z_i, z_j)] \cdot P(\mathbf{w}_i^j | a_n)$ maximiert. Zur Ermittlung der wahrscheinlichsten Folge \mathbf{w}_i^j wird die *Dynamische Programmierung* verwendet. Für das Attribut a_n ergibt sich somit der Zustandsübergang $(z_i, z_j, \mathbf{w}_i^j, a_n, w_{WK})$ in \mathcal{A}_{WK} . Aus der ermittelten Transitionsmenge E_{WK} zu allen Knotenpaaren wird der Graph \mathcal{A}_{WK} erstellt.

Anschließend wird die wahrscheinlichste Attributfolge $\hat{\mathbf{a}}$ dekodiert. Die Entscheidung ergibt sich aus den kombinierten Bewertungen vom semantischen Sprachmodell für Attributfolgen und den dazugehörigen Gesamtgewichten durchgehender Pfade des Automaten \mathcal{A}_{WK} . Die wahrscheinlichste Konzeptmarkierungssequenz (\mathbf{w}, \mathbf{a}) wird für ein abschließendes *slot-filling* verwendet [CHG⁺16].

Anmerkungen zur Kombination von Spracherkennung und Interpretation

In der Praxis wird üblicherweise eine sequenzielle Dekodierung vorgenommen und nur die beste Hypothese $\hat{\mathbf{w}}$ für die Interpretation verwendet [HDR⁺11, DSTHT12]. Diese Vorgehensweise liefert nicht immer optimale Resultate. Gerade falsch erkannte Schlüsselwörter sind für eine Konzeptzuordnung nachteilig. Aus diesem Grund ist es sinnvoll, endgültige Entscheidungen möglichst spät durchzuführen. Bewertete Hypothesen können beispielsweise als *N-besten Liste* oder als Worthypothesengraph vom Sprachanalysemodul an das Interpretationsmodell weitergeleitet werden [JHJ98]. So werden z. B. in [DSTHT12] Worthypothesengraphen (für die Konzeptbeschriftung mittels *Conditional Random Fields* und *Maximum-Entropie Markovmodelle* verwendet.

Eine weitere Möglichkeit stellt die einheitliche Verwendung von endlichen Transduktoren dar. So wurde in [CLM⁺01] eine integrierte Spracherkennung und -übersetzung untersucht. Für die automatische Zuordnung von Konzeptbeschriftungen konnte in [SRBN06] für erste Untersuchungsergebnisse eine Verbesserung der *Concept Error Rate* (CER) festgestellt werden.

Durch den Klassifikationsprozess eines Spracherkenners erfolgt die Zuordnung möglicher Sprachzeichenfolgen zu einer akustischen Beobachtung $\bar{\mathbf{o}}$. Zu jeder einzelnen Wortfolge können möglicherweise durch den Interpretationsprozess wieder unterschiedliche Konzeptmarkierungsfolgen erzeugt werden. Bei Verwendung des generativen Ansatzes wird die Wahrscheinlichkeit des Auftretens einer Konzeptmarkierungsfolge \mathbf{t} zu den akustischen Merkmalen $\bar{\mathbf{o}}$ durch

$$P(\mathbf{t}|\bar{\mathbf{o}}) = \frac{1}{P(\bar{\mathbf{o}})} \left(\sum_{\mathbf{w}} P(\bar{\mathbf{o}}|\mathbf{w}, \mathbf{t}) \cdot P(\mathbf{w}, \mathbf{t}) \right) \quad (3.20)$$

bestimmt [Mor07, HDR⁺11]. Dabei wird letztendlich diejenige Konzeptmarkierungsfolge ausgewählt, welche die höchsten A-posteriori Wahrscheinlichkeit $P(\mathbf{t}|\bar{\mathbf{o}})$ aufweist. Die optimale Hypothese $\hat{\mathbf{t}}$ ergibt sich mit Formel 3.20 und dem konstanten Wert $P(\bar{\mathbf{o}})$ durch

$$\begin{aligned} \hat{\mathbf{t}} &= \arg \max_{\mathbf{t}} P(\mathbf{t}|\bar{\mathbf{o}}) \\ &= \arg \max_{\mathbf{t}} \sum_{\mathbf{w}} P(\bar{\mathbf{o}}|\mathbf{w}, \mathbf{t}) \cdot P(\mathbf{w}, \mathbf{t}) = \arg \max_{\mathbf{w}, \mathbf{t}} P(\bar{\mathbf{o}}|\mathbf{w}) \cdot P(\mathbf{w}, \mathbf{t}). \end{aligned} \quad (3.21)$$

Die Wahrscheinlichkeit $P(\bar{\mathbf{o}}|\mathbf{w})$ wird durch die akustischen Modelle im Spracherkennung geschätzt. Zur Bestimmung der Verbundwahrscheinlichkeit $P(\mathbf{w}, \mathbf{t})$ wird hier das *konzeptuelle Sprachmodell* verwendet.

3.2 Artikulation von Sprachzeichenfolgen

Die automatische Produktion eines natürlichsprachlichen Textes aus einer vorliegenden Bedeutungsrepräsentation ist Gegenstand der Artikulation. Die zu übermittelnden Informationen müssen hierbei lexikalisiert und linearisiert werden. Die Auswahl einer Informationsstruktur und deren Artikulation wird auch als *Text- oder Sprachgenerierung* (*natural language generation*) bezeichnet. Die erzeugte Wortfolge kann als lesbarer Text ausgegeben oder zur Generierung einer akustischen Sprachäußerung verwendet werden. Die Sprachsynthese ist beispielsweise in technischen Anwendungen notwendig, welche die Bedürfnisse von blinden oder sehbehinderten Menschen berücksichtigen möchten oder in Systemen, die keine Displays zur visuellen Anzeige verwenden. Weiterhin ist eine Sprachgenerierung besonders in den Anwendungen sinnvoll, in denen auch der Benutzer natürlichsprachliche Systemanfragen stellen kann.

Artikulation und Synthese sind in einem Sprachproduktionssystem eng miteinander verbunden und werden nicht immer als zwei separate Teile betrachtet. Die bei der Textgenerierung auszuwählenden syntaktischen Ausdrücke sind vom Kommunikationsziel, der Dialoghistorie sowie vom Nutzer abhängig. Die als *Eingabe* verwendeten Daten und Wissensstrukturen einer Domäne können z. B. Datenbankeinträge, Repräsentationen von Weltwissen oder auch personalisierte Informationen über den Benutzer (Benutzermodelle) sein. Typischerweise werden durch den Sprachgenerierungsprozess zwei wesentliche Funktionen realisiert:

1. Auswahl der Bedeutung, die artikuliert werden soll (*Inhaltsfestlegung*)
2. formbestimmende Generierung einer entsprechenden Wortfolge zu der ausgewählten Informationsstruktur (*Artikulation*)

Die Auswahl einer zu artikulierenden Bedeutungsstruktur ist Teil der Makroplanung und orientiert sich an dem Handlungsziel eines intelligenten Sprachdialogsystems. Diese *Inhaltsfestlegung* ist ein Aufgabenbereich der *Verhaltenssteuerung* bzw. des *Dialogmanagers* in einem Sprachdialogsystem und wird hier nicht näher betrachtet.

3.2.1 Artikulation mittels vorgefertigter Textschablonen

Die einfachste Möglichkeit der Artikulation ist die direkte Umsetzung mittels vorgefertigter Textteile oder die kontextsensitive Verwendung von Textschablonen. Für Sprachdialogsysteme stellt diese „flache Generierung“ die zugleich am häufigsten verwendete Methode dar. Schon beim Dialogdesign werden die zu artikulierenden Sprachäußerungen festgelegt. Je nach Art des Dialogschritts wird dabei schon ein Großteil der Sprachzeichen für die zu generierende Wortfolge spezifiziert. Bei der Artikulation sind dann

nur noch die individuell ermittelten Werte in vorbereitete Text-Schablonen einzusetzen. Zum Beispiel sind in dem vorgefertigten Fragesatz:

Wann möchten Sie von <dep_city> nach <arr_city> fahren?

die enthaltenen Platzhalter für Abfahrts- und Zielort mit den zuvor bestimmten Datenwerten zu substituieren. Ein wesentlicher Nachteil besteht in der Inflexibilität dieses Verfahrens. Dies betrifft konkret die Variabilität des Artikulationsergebnisses und die Anpassungsmöglichkeiten auf neue Anwendungsgebiete. Einen allgemeinen generischen Ansatz für die Spracherzeugung mit Text-Schablonen findet man zum Beispiel in [TKP⁺01].

3.2.2 Standardmodell der regelbasierten Sprachgenerierung

Für die „tiefe Generierung“ von wohlgeformten Wortfolgen mit Hilfe von Grammatiken findet man in der Literatur verschiedene Planungsverfahren. Eine Beschreibung des typischen Textgenerierungsprozesses ist beispielsweise durch das in [RD00] vorgestellte *Konsensmodell* (siehe Abb. 3.13) gegeben. Zu der Inhaltsfestlegung gehört hierbei

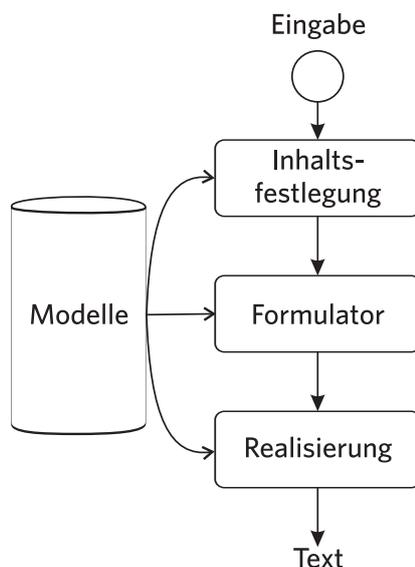


Abb. 3.13: Prinzipielle Struktur des Konsensmodells

u. a. auch eine Grobstrukturierung des Textes, die sogenannte *Inhaltsorganisation*. Anschließend überführt der *Formulator* die präverbale Bedeutungsrepräsentation in eine sogenannte *Oberflächenformspezifikation*. Für die genauere Satzplanung ist gegebenenfalls eine *Aggregation* (Zusammenfassung) notwendig. Als Beispiel für eine mögliche *Aggregation* dient die längere Wortfolge „*Peter trinkt Bier. Steffen trinkt Bier.*“. Diese kann z. B. auch durch den Satz „*Peter und Steffen trinken Bier.*“ zusammengefasst werden. Die Bedeutungsstruktur enthält in der Regel nur semantische Primitiva. Zu einem Konzept sind typischerweise jedoch mehrere sprachliche Ausdrücke referenziert.

Deshalb muss im *Formulator* auch eine geeignete Lexikalisierung vorgenommen werden. Im Anschluss wird u. a. durch Festlegung der Syntax und der Flexion der Lexeme eine grammatisch stimmige Äußerungspartitur erstellt. Für die Zuordnung der syntaktischen Struktur einer Sprachäußerung werden zum Teil komplexe Grammatiken verwendet. Exemplarisch können hierfür die Lexical Functional Grammar (LFG), Combinatory Categorical Grammar (CCG), Head-Driven Phrase Structure Grammar (HPSG) und die Tree Adjoining Grammar (TAG) aufgeführt werden [GWG11]. Durch den Einsatz von Grammatiken ist die Berücksichtigung von entsprechenden semantischen, syntaktischen und morphologischen Informationen bei der Artikulation möglich. Die abschließende *Realisierungskomponente* erzeugt einen lesbaren und ggf. formatierten Text in dem gewünschten Ausgabeformat.

Ein System, welches ein solches mehrstufiges Generierungsverfahren nutzt, ist das in [WBB⁺10] vorgestellte multilinguale Informationssystem MARQUIS. Dieses informiert den Anwender über aktuelle Schadstoffwerte in der Luft und über die prognostizierte Luftverschmutzung. In den sprachverarbeitenden Systemen werden zumeist unterschiedliche Repräsentationsformen für die semantischen Daten verwendet. Auch die Zwischenergebnisse der implementierten „tiefen“ planungsbasierten Generierungsverfahren variieren. Weiterführende Informationen findet man u. a. in [RD00, Bus03, Sch06, Rei10, CEE10]

3.2.3 Verwendung statistischer Verfahren zur Artikulation

Datenbasierte Methoden können natürlich auch zur Artikulation von Sprachzeichenfolgen verwendet werden. Die derzeitigen Verfahren liefern nicht immer wohlgeformte sprachliche Äußerungen. Oft werden in den Veröffentlichungen nur bestimmte Teilspekte des kompletten Sprachgenerierungsprozess betrachtet.

Grob kann zwischen zwei verschiedenen Klassen von datenbasierten Ansätzen zur Artikulation unterschieden werden:

- Regelbasierte Generierungsverfahren mit statistischer Bewertung,
- Stochastische Sprachgenerierung.

Regelbasierte Generierungsverfahren mit statistischer Bewertung

Diese Verfahren basieren auf dem Prinzip der *Übergenerierung*. Dabei wird mittels manuell erstellter Grammatiken eine Vielzahl möglicher Wortfolgen erzeugt. Anschließend erfolgt mit Hilfe von Referenzdaten die Erstellung einer Rangfolge. Die erste Methode hierfür wurde in [KH95] vorgestellt. Ein sogenannter regelbasierter Symbolgenerator erzeugt entsprechende Wortfolgen. Dieser Symbolgenerator übernimmt hierbei die

Funktion der Lexikalisierung und Linearisierung. Die Auswahl geeigneter Sequenzen von Sprachzeichen aus einem Worthypothesengraph erfolgt, wie auch in dieser Arbeit, durch ein *N-Gramm-Sprachmodell* (vgl. Kapitel 2.4). Informationen zu Weiterentwicklungen dieses Systems findet man beispielsweise in [LK98] und [Lan00].

Ein vergleichbares Verfahren wird in [BR00] beschrieben. Als Eingabe liegt ein Ableitungsbaum vor, welcher nur die zu generierenden Wörter beinhaltet. Zur Erzeugung eines Worthypothesengraphen werden hierbei allerdings stochastische Baummodelle verwendet.

In [ES05] wird ausschließlich die Bewertung erzeugter Sprachäußerungen untersucht. Es erfolgt eine vergleichende Analyse der Bewertungsergebnisse durch ein *N-Gramm-Sprachmodell* und ein *Maximum-Entropie-Modell*. Weiterhin gibt es auch Ansätze, die für die Auswahl einer Wortfolge Persönlichkeitsmerkmale [IBO06] oder die geschätzte Natürlichkeit einer synthetisierten Sprachäußerung [CM06] verwenden.

Statistische Sprachgenerierung

Die in diesem Abschnitt vorgestellten Verfahren übersetzen eine Semantikrepräsentation in eine natürliche Zielsprache. Die Modelle für die Generierung werden hierbei automatisch mit geeigneten Trainingsdaten erstellt.

Zum Beispiel werden in [OR00] die Wortfolgen durch stochastische Sprachmodelle erzeugt. Zum Training sind reale Sprachdialoge (Domäne: Reisereservierungen) verwendet worden. Für die Artikulation werden semantische Rahmen mit Attribut-Wert-Paaren vorgegeben. Zu jeder Sprachäußerungsklasse generiert ein individuelles Sprachmodell zufällig Wortsequenzen, die anschließend nach verschiedenen Kriterien bewertet werden. Die am besten geeignete Wortfolge wird ausgewählt und die darin enthaltenen Attribute (Wortklassen) durch die dazugehörigen Werte des semantischen Rahmens ersetzt. Für dieses System ist eine semantisch annotierte Lernstichprobe nötig, bei der zusätzlich jede Sprachäußerungen thematisch zu klassifizieren ist.

Vier verschiedene Konzepte für stochastische Modelle zur *Oberflächenrealisierung* werden in [Rat02] vorgeschlagen. Ausgangspunkt ist auch hier eine gegebene Menge von Attribut-Wert-Paaren. Ziel dieser Verfahren ist die Bestimmung der Reihenfolge der Attribute und die anschließende Lexikalisierung. Der Trainingskorpus besteht aus einer Menge von Sprachäußerungen (Genre: Fluginformation), bei denen bestimmte Wörter durch das dazugehörige Attribut ersetzt wurden. Die Bestimmung der wahrscheinlichsten Sprachzeichenfolge erfolgt hier u. a. durch eine *Maximum-Entropie-Klassifikation*.

Statistische maschinelle Übersetzungsverfahren werden in [WM07] zur Textgenerierung untersucht. Neben einer *phrasenbasierten Übersetzung* erfolgt hier die Anwendung eines *syntaxbasierten maschinellen Übersetzungsverfahrens*. Grundlage dafür ist ein *inverser probabilistischer semantischer Parser*, welcher auf einer *kontextfreien Grammatik* basiert.

Zur Bestimmung der wahrscheinlichsten Sequenz, zu einem linearisierten semantischen Baum, wird neben dem Übersetzungsmodell auch ein *N-Gramm-Sprachmodell* der Zielsprache genutzt [WM07].

In [LNL09] werden hybride Bäume verwendet, welche semantische Einheiten und dazugehörige Phrasen der natürlichen Sprache beinhalten. Der Trainingskorpus ist hierbei jeweils aus einer Menge von Paaren zusammengesetzt, bestehend aus einem semantischen Baum und der dazugehörigen Paraphrase der Zielsprache.

Der statistische Sprachgenerator BAGEL [MGJ⁺10, MY14] verwendet *Dynamische Bayes'schen Netze*. Für die linearisierte Darstellung einer semantischen Baumstruktur werden *Stapelspeicher* eingesetzt. Jedem Blatt des Baumes ist genau ein *Stapelspeicher* zugeordnet. Diese Darstellungsform von semantischen Einheiten ist ähnlich zum *Hidden Vector State Modell* [HY05], welches für die semantische Interpretation verwendet wird (siehe auch Abb. 3.11). Der Inhalt dieses Speichers ergibt sich aus den übergeordneten Knotenbeschriftungen. Die Bezeichnung des Handlungsziels bzw. des Dialogakts, welches die Wurzel des Baums bildet, ist das unterste Element des Speichers. Für jede Sprachäußerung der Lernstichprobe ist manuell eine Zuordnung des semantischen Konzepts (*Stapelspeicher*) zu einem Wort bzw. einer Phrase getroffen worden. Dabei wird zwischen obligatorischen Speichern und zusätzlichen (semantisch irrelevanten) Speichern (*filler stacks*) unterschieden. Bei der Sprachgenerierung muss nun zu einer gegebenen ungeordneten Menge von *Stapelspeichern* die passendste Sprachzeichenfolge gefunden werden. Hierfür werden sogenannte *factored language models* [BK03] verwendet. Dieser Ansatz ist äquivalent zu *Dynamischen Bayes'schen Netzen*, in welchen die Wahrscheinlichkeiten der Kindknoten aus den verschiedenen Elternknoten geschätzt wird [MY14]. *Factored language models* ermöglichen allgemein die Berücksichtigung von zusätzlichen Informationen, wie z. B. semantischen Attributen oder Wortarten. Das *i*-te Wort w_i wird durch K Merkmale beschrieben. Dies wird durch die Gleichung

$$w_i = \{m_i^1, m_i^2, \dots, m_i^K\} \quad (3.22)$$

ausgedrückt [BK03]. Dabei kann ein Merkmal auch das Wort w_i selbst sein. Die Dekodierung in dem System BAGEL erfolgt in drei aufeinanderfolgenden Schritten. Es werden jeweils die n -besten Sequenzen aus der Sprachmodellbewertung von den:

1. obligatorischen Stapelspeichersequenzen,
2. Folgen von kompletten Speichersequenzen (nach der Einfügung einer begrenzten Anzahl von *filler stacks*) und
3. realisierten Phrasenfolgen

ausgewählt und weiterverwendet [MY14].

Der in [WGK⁺15] verwendete Ansatz zur Textgenerierung basiert hingegen auf *rekurrenten neuronalen Netzen*. Diese werden für eine Übergenerierung und zur Reihenfolgenbestimmung verwendet. Das generierende Sprachmodell wird mit Paaren von Dialogakttyp und dazugehöriger Sprachäußerung trainiert. Dabei werden zuvor in der Lernstichprobe die semantisch relevanten Wörter durch die assoziierte Attributbezeichnung ersetzt.

Zusätzlich sind z. B. in [PE05, FW08] Verfahren zur Wortfolgenerierung beschrieben. Bei diesen Systemen werden lediglich variable Parameter der manuell erstellten Generierungsmodelle anhand von Trainingsdaten optimiert.

Kapitel 4

Äußerungs-Bedeutungs-Transduktoren

In diesem Kapitel erfolgt die Darstellung der wesentlichen Grundlagen zur semantischen Sprachverarbeitung mit endlichen *Äußerungs-Bedeutungs-Transduktoren* (*utterance meaning transducers*). Diese endlichen Automaten werden für den bidirektionalen Übersetzungsprozess zwischen Sprachzeichenfolgen und semantischen Zeichenketten eingesetzt.

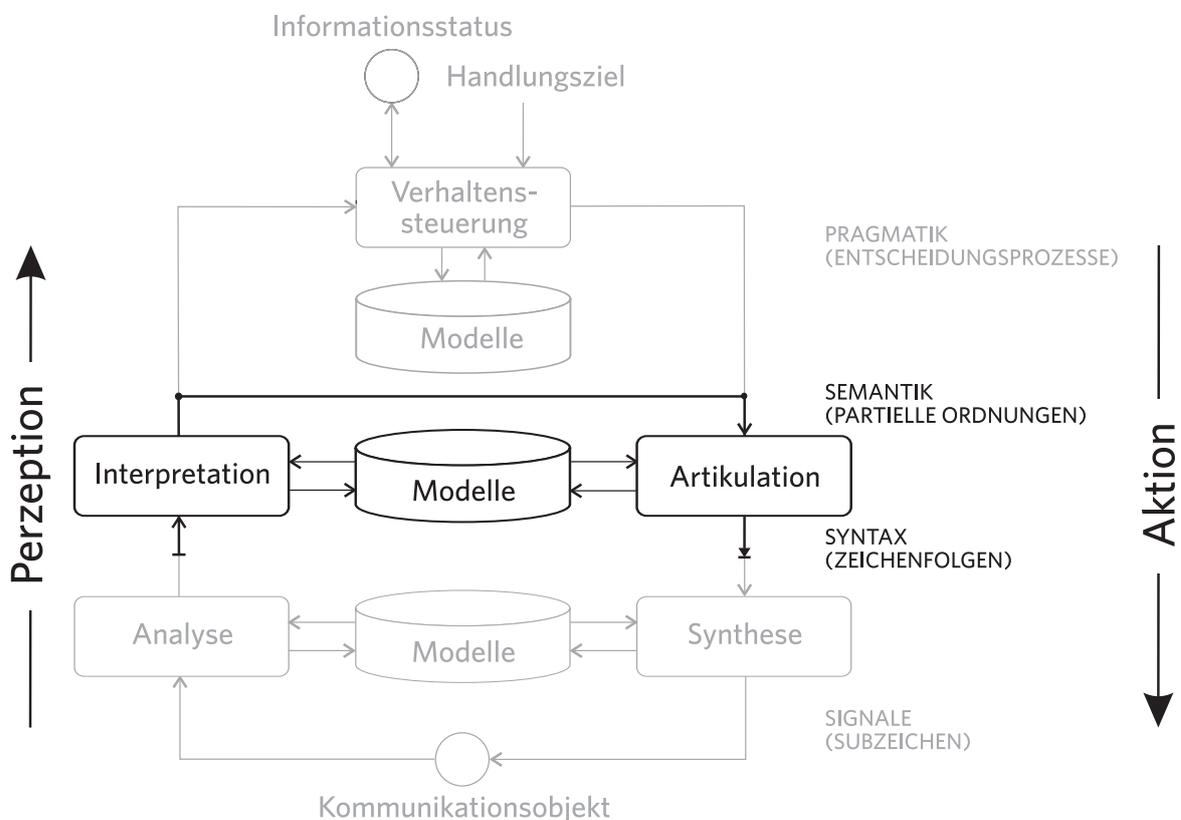


Abb. 4.1: System zur gemeinsamen Interpretation und Artikulation mittels Äußerungs-Bedeutungs-Transduktoren (Übersicht) nach [HW15]

Die Interpretation von sprachlichem Verhalten und die Artikulation einer Bedeutungsstruktur bildet die Schnittstelle zwischen der syntaktischen und der semantischen Ebene eines hierarchischen Sprachdialogsystems. In der Abb. 4.1 ist ein vereinfachtes Strukturbild des verwendeten Systemmodells dargestellt. Die für diese Arbeit relevanten Funktionsblöcke sind darin hervorgehoben. In der erstellten Beispielanwendung wird anstelle einer semantischen Weiterverarbeitung die durch die Interpretation erzeugte Bedeutungsrepräsentation direkt wieder artikuliert.

4.1 Merkmal-Werte-Relationen als Bedeutungsrepräsentation

In dem für diese Arbeit verwendeten Modell eines intelligenten hierarchischen Systems sind für die semantische und pragmatische Ebene einheitliche Datenstrukturen vorgesehen. Für die Repräsentation von außersprachlichem Wissen werden sogenannte *Merkmal-Werte-Relationen* (MWR) [HKL+09] benutzt. Eine *Merkmal-Werte-Relation* ist ein azyklischer, beschrifteter gerichteter Graph

$$G = (V, <, l). \quad (4.1)$$

Dieser Graph besteht dabei aus einer Menge V von Knoten (*vertices*) und der Kantenmenge $< \subset V \times V$. Die binäre Relation („kleiner als“-Beziehung) über V ist irreflexiv und transitiv. In der Formel 4.1 entspricht $l : V \rightarrow L$ der Etikettierungsfunktion. Diese ordnet den Knoten Beschriftungen aus einer endlichen Etikettenmenge L zu [WHK+12, LHW14]. Als Beschriftungen dienen hierbei die Bezeichnungen semantischer Kategorien sowie Datenwerte. *Merkmal-Werte-Relationen*, denen Wahrscheinlichkeitsmaße zugeordnet sind, werden als *gewichtete Merkmal-Werte-Relationen* $G = (V, <, l, w)$ mit $w : V \rightarrow \mathbb{K}$ bezeichnet. *Merkmal-Werte-Relationen* können als *beschriftete partielle Ordnungen* aufgefasst werden. Auch die Bezeichnungen *beschriftete partielle Wörter* über L , *beschriftete Halbordnungen* oder *beschriftete, partiell geordnete Multimengen* sind gebräuchlich.

Ein Beispiel für eine MWR ist durch Abb. 4.2 gegeben. Aufgrund der gerichteten Kanten und der Azyklizität entspricht die Darstellung der MWR einem gewurzelten Baum. Ein Merkmal kann als Hierarchie von semantischen Kategorien (Entitätstypen und Attribute) aufgefasst werden. Zum Beispiel gehört zu dem Objekt „Person“ die untergeordnete semantische Kategorie „Name“. Mit dieser sind wiederum die Attribute „Vorname“ und „Familienname“ assoziiert. Diesen Merkmalen können konkrete Datenwerte zugeordnet werden, welche sich aus der analysierten Sprachzeichenfolge ergeben.

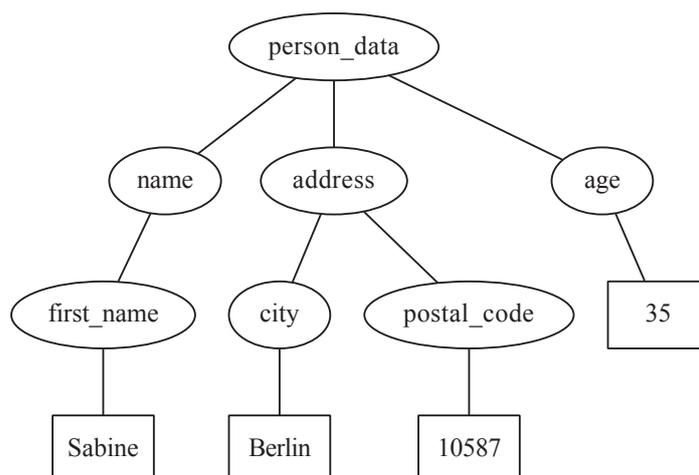


Abb. 4.2: Ungewichtete Merkmal-Werte-Relation

Sind zu Merkmalen entsprechende Datenwerte bekannt, so sind diese in den Blättern eingetragen. In Abb. 4.2 sind die Wertknoten als Rechteck dargestellt. Auf diese Weise können einzelne Bedeutungseinheiten in der Baumstruktur identifiziert werden.

Merkmal-Werte-Relationen dienen nicht nur zur Repräsentation der Ergebnisse des semantischen Interpretationsvorgangs, sondern z. B. auch zur Darstellung des Informationsstatus sowie des Weltmodells [KW11, WHK⁺12]. Zudem werden diese Datenstrukturen auch für die Artikulation einer ausgewählten Systemaktion benötigt.

In dem Dialogsystemmodell kann zur Verwaltung und Identifikation von Datenobjekten ein relationales Datenbanksystem verwendet werden. Solche Datenbanken können durch Entity-Relationship-Modelle [Che76] beschrieben werden. Wie in [HKL⁺09] gezeigt, können *Merkmal-Werte-Relationen* direkt aus Entity-Relationship-Modellen erzeugt werden. Dabei beschreiben die automatisch generierten Datenstrukturen die bestehenden Beziehungen zwischen Entitätstypen. Weiterhin können *Merkmal-Werte-Relationen* auch zur Manipulation von relationalen Datenbanken verwendet werden. Dafür muss die MWR allerdings um eine ID-Schicht erweitert werden [HKL⁺09]. Für die weiteren Ausführungen in dieser Arbeit ist die Berücksichtigung der Verbindung von Werten zur Datenbank jedoch nicht relevant. In [Wir12] wurde gezeigt, dass semantische Daten nichtsequenziell sind. Im Normalfall spielt die Reihenfolge der artikulierten Bedeutungseinheiten einer elementaren *Merkmal-Werte-Relation* für das weitere Sprachverstehen keine Rolle. Eine Totalordnung oder eine ungeordnete Menge von *Konzepten*, wie sie in einigen Systemen als Ergebnis der Interpretation vorliegt, kann die bestehenden semantischen Relationen nur unzureichend abbilden. Partielle Wörter haben die Eigenschaft auch nebenläufige Ereignisse darzustellen zu können. Diese Notationsform ermöglicht somit eine sinnvolle Strukturierung von semantischen Einheiten. Es besteht hierbei sowohl die Möglichkeit der Darstellung einer hierarchischen Ordnung von Merkmalen als auch eine Repräsentation von bestehenden Nebenordnungsrelationen.

Gewichtete Merkmal-Werte-Relationen ermöglichen zudem die Berücksichtigung von unsicherem Wissen. Dies alles sind wesentliche Argumente für die Verwendung von partiellen Wörtern zur Semantikrepräsentation und auch zur kompletten Bedeutungsverarbeitung.

4.2 Semantische Grammatiken

Ausgangspunkt ist der in [WL13] vorgestellte Ansatz für ein bedeutungsorientiertes Sprachmodell. Die darin vorgenommene Beschreibung der Bedeutung einer sprachlichen Äußerung basiert auf den von BURRHUS FREDERIC SKINNER veröffentlichten Analysen verbalen Verhaltens [Ski57]. Dies stellt die Grundlage für die Modellierung und den Einsatz von semantischen Grammatiken in unserem Systemmodell dar.

4.2.1 Konzept des bedeutungsorientierten Sprachmodells

Zur Ermittlung der kommunikativen Bedeutung M von verbalem Verhalten (*behavior* B) ist die Analyse des bestehenden Kontextes nötig. Dazu gehören die Vorgeschichte (*antecedents* A) und die beabsichtigten Auswirkungen (*consequences* C). Analog zum Shannon'schen unidirektionalen Kommunikationsmodell [Sha48], mit den Komponenten Sender, Kanal und Empfänger, kann ein Kommunikationsprozess durch das folgende abstrakte ABC-Schema dargestellt werden:

Antecedents \rightarrow **Behavior** \rightarrow **Consequences**

Dieselbe Sprachzeichenfolge kann zu unterschiedlichen Bedeutungen M_i gehören. Wenn man z. B. alleinig die sprachliche Äußerung „*Nein, ich möchte das nicht.*“ betrachtet, kann die intendierte Absicht des Sprechers nicht ermittelt werden. Das Wort „das“ bezieht sich hierbei auf bestimmte Vorbedingungen, welche hier unbekannt sind. Sinnvoll ist demnach die Verwendung situationsabhängiger Parameter für die Ermittlung der Bedeutung eines natürlichsprachlichen Ausdrucks begrenzter Länge. Dazu dient hier eine Menge kompatibler *semantischer Anker* $\{a_i\} \in A$ und $\{c_j\} \in C$, welche bestimmte Sachverhalte repräsentieren. *Semantische Anker* stellen somit eine subjektive Referenz zu einem subjektiv wahrgenommenen Aspekt der Realität her [WL13].

Die Bedeutung einer Sprachhandlung ergibt sich dabei durch

$$M = \{\{a_i\}, \{c_j\}\} \quad \text{mit } \{a_i\} \in A \text{ und } \{c_j\} \in C. \quad (4.2)$$

Nun kann ein Mensch in einer vergleichbaren Kommunikationssituation unterschiedliches verbales Verhalten B zeigen. Es ist nicht möglich alle Einflussfaktoren durch die Wertemengen A und C zu erfassen und formal zu beschreiben. Ein Zusammenhang zwischen der Sprachäußerung und dem Gemeinten kann somit nur empirisch hergestellt werden. Aus diesem Grund ist es sinnvoll, das beobachtbare verbale Verhalten als Zufallsgröße zu betrachten.

Die endliche Menge möglicher Sprachäußerungen zu denselben *semantischen Ankern* wird mit $\mathcal{U}(M)$ bezeichnet. Wenn die relativen Auftretenshäufigkeiten von Sprachäußerungen $\mathbf{x}_i \in \mathcal{U}(M)$ bekannt sind, so kann zu möglichen sprachlichen Verhalten die bedingte Wahrscheinlichkeitsverteilung $P(\mathbf{x}|\mathcal{U}(M))$ angegeben werden. Sind die notwendigen Trainingsdaten verfügbar, so ist auf diesen Weg die Erstellung eines elementaren stochastischen Sprachmodells $\mathcal{L}(M)$ möglich.

Je mehr *semantische Anker* gegeben sind, umso konkreter wird die Kommunikationssituation und auch die Bedeutung M definiert. Tendenziell wird mit steigender Anzahl *semantischer Anker* die Mächtigkeit der Menge möglicher Äußerungen $|\mathcal{U}(M)|$ kleiner. Auch wenn der Ausschnitt der zu modellierenden Welt für ein technisches System eingeschränkt ist, so sind oft sehr viele unterschiedliche Absichten des Zeichenbenutzers interpretierbar. Wahrscheinlich ist ein Großteil möglicher Äußerungen in einer Sprachdatensammlung nicht enthalten.

Aus diesem Grund ist es für die Modellierung eines semantischen Sprachmodells sinnvoll, bestimmte *semantische Anker* durch *Platzhalter* zu ersetzen. Dies sind unspezifizierte Verweise auf eine Menge möglicher *semantischer Anker*. Wird nun zu einer konkreten Bedeutung M mindestens ein *semantischer Anker* durch einen *Platzhalter* substituiert, so erhält man ein *semantisches Schema* S . Eine Wortfolge, welche mit der Bedeutung M assoziiert ist, stellt andersherum eine *Spezifikation* des *semantischen Schemas* dar. Für ein bedeutungsorientiertes Sprachmodell muss nun eine Menge von Grammatiken zu möglichen *semantischen Schemata* konstruiert werden. Für die Spezifikationen von Grammatiken mit *Platzhaltern* sind *domänenspezifisches* sowie *linguistisches Wissen* nötig.

Diese Grammatiken können als Grundlage für die Modelle zur Interpretation und Artikulation in einem Dialogsystem verwendet werden [WL13, Rit13].

4.2.2 Hierarchie von Grammatiken

Ausgehend von den Ausführungen im Kapitel 4.2.1 kann eine hierarchische Ordnung von bedeutungsorientierten Sprachmodellen angegeben werden. Sprachverarbeitende technische Anwendungen, wie z. B. natürlichsprachliche Dialogsysteme, haben zumeist einen aufgabenbezogenen Gegenstandsbereich. In dieser Domäne D ist eine endliche Menge sprachlicher Realisierungen $\mathcal{U}(D)$ denkbar, welche durch eine reguläre Grammatik (vgl. Kapitel 2.4) beschrieben werden können. Die *semantischen Anker* zu der *lokalen Grammatik* grenzen dabei den zu modellierenden Weltausschnitt ein. Eine *mikrolokale Grammatik* definiert die Menge möglicher Sprachäußerungen $\mathcal{U}(S)$. Sie ist eine Komponente der *lokalen Grammatik* und gehört zu einem *semantischen Schema* S . Zu einer *mikrolokalen Grammatik* wird in dieser Arbeit ein endlicher *Äußerungs-Bedeutungs-Transduktor* zugeordnet, welcher die generierte Sprache der Grammatik akzeptiert. Eine *lokale Grammatik* $\mathcal{G}(D)$ kann mit

$$\mathcal{G}(D) = \bigoplus_i \mathcal{G}(S_i) \quad (4.3)$$

als die Vereinigung aller mikrolokalen Grammatiken $\mathcal{G}(S_i)$ zu der Domäne D betrachtet werden. Die Grammatik welche nur die Menge von Sprachzeichenfolgen $\mathcal{U}(M)$ zu einer gegebenen Bedeutung M generiert, wird als *Elementargrammatik* bezeichnet. Folglich können nun neben der Rangfolge der aufgeführten Grammatiken (*lokale Grammatik* - *mikrolokale Grammatik* - *Elementargrammatik*) auch die dazugehörigen Mengenbeziehungen $\mathcal{U}(D) \supseteq \mathcal{U}(S) \supseteq \mathcal{U}(M)$ angegeben werden [WL13, WW14].

4.2.3 Bezug zur Interpretation in einem Sprachdialogsystem

Zur Veranschaulichung des gegebenen Äußerungskontextes von verbalem Verhalten eines Benutzers dient die Betrachtung eines Dialogschritts. In sogenannten agentenbasierten Dialogsystemen werden Entscheidungen dynamisch und zielorientiert getroffen. Mit der Auswahl einer geeigneten Kommunikationshandlung h versucht der Agent das Kommunikationsobjekt zielgerichtet zu beeinflussen. Der innere Zustand des Benutzers z ist selbst nicht zugänglich. Somit ist auch die tatsächliche Wirkung eines Aktuatorsignals auf die Umwelt unbekannt. Die assoziierten Ziele des Kommunikationsobjektes und dessen möglicher Zustand müssen daher aus dem beobachteten Nutzerverhalten ermittelt werden.

Vereinfacht können die beiden Handlungen vom Agenten und Benutzer während eines Dialogturns durch folgende Schritte beschrieben werden:

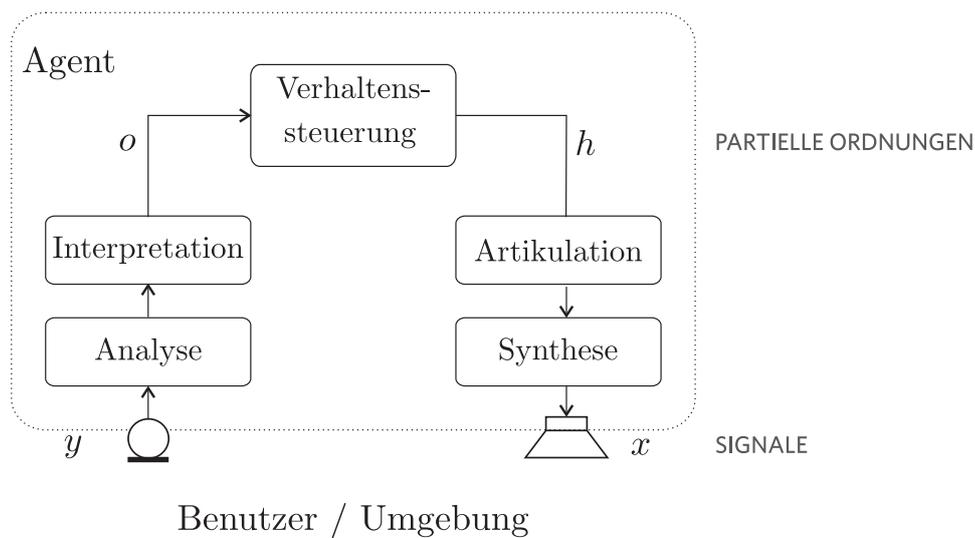


Abb. 4.3: Sequenzieller Ablauf eines Dialogschritts - visualisiert am Blockschaltbild des Dialogsystems

1. Zielorientierte Auswahl einer zu artikulierenden Aktion h anhand einer erlernten Strategie und des aktuellen geschätzten Zustands \tilde{z} des Kommunikationsobjekts.
2. Emittiertes Aktorsignal x führt ggf. zu einer Zustandsänderung der Umwelt $z' = f(z, x)$.
3. Erfassung des (möglicherweise gestörten) Signals y durch Sensorik.
4. Perzeption des Messsignals liefert geschätzte Absicht o des Benutzers.

Zur Veranschaulichung der allgemeinen Beschreibung eines Interpretationsvorgangs dient die Abb. 4.3. Eine Ermittlung des neuen Zustands \tilde{z}' der Umwelt und die Auswahl einer optimalen Folgehandlung ($\tilde{z}' \rightarrow h'$) muss dabei unter Unsicherheit getroffen werden. Die stochastische Modellierung eines solchen Dialogmanagements ist z. B. durch *Markov'sche Entscheidungsprozesse* [KLC98, LPE00, YGTW13] möglich.

Verfügbare Vorbedingungen

Zur Bestimmung der Bedeutungsträger o zu dem verbalen Verhalten dient dem Interpretieren u. a. die Menge an Vorbedingungen (*antecedents*). Die zu interpretierende Sprachäußerung selbst ist das Bindeglied zwischen der Vorgeschichte und der beabsichtigten Konsequenz.

Zu den *semantischen Anker* A gehört z. B. die Dialoghistorie. In unserem Systemmodell sind ermittelte *Merkmal-Werte-Relationen* (vgl. Kapitel 4.1) aus vorherigen Dialogschritten im Informationsstatus enthalten. Zudem gehört auch der im letzten Dialogschritt geschätzte Zustand der Umwelt \tilde{z}_t sowie indirekt die zuvor ausgewählte Handlung h zu den Vorbedingungen A . Die zum diskreten Zeitpunkt t ermittelten Konsequenzen C stellen somit eine Teilmenge der *antecedents* A' dar. Diese dienen zur Ermittlung der Intention des Dialogakts zum darauffolgenden Zeitpunkt t' . In einigen

Anwendungen ist möglicherweise auch die Verwendung eines Benutzermodells sinnvoll. *Semantische Anker* zu einem Benutzerprofil sind z. B. abgespeicherte Präferenzen und weiteres benutzerspezifisches Wissen aus früheren Dialogen.

Zu einer Menge *semantischer Anker* $\{a_j\} \subseteq A$ kann für den aktuellen Perzeptionsprozess eine endliche Menge möglicher Absichten des Kommunikationsobjekts abgeleitet werden. Diese orientieren sich z. B. an dem Handlungsziel des Benutzers und der momentan vorherrschenden Dialoginitiative. Im Idealfall liegen dem Agenten für mögliche Klassen von Dialogakten *semantische Schemata* S_i sowie eine geschätzte bedingte Wahrscheinlichkeitsverteilung $P(S_i|\{a_j\})$ vor. Die zugeordneten *mikrolokalen Grammatiken* können auf diese Weise für den Interpretationsprozess initial unterschiedlich gewichtet werden. Alternativ ist die Aktivierung genau derjenigen Transduktoren möglich, welche zum aktuell bestehenden Kontext passen. Diese Möglichkeit wurde bereits in [WL13] angegeben. In der Regel geht man dabei von einem kooperativen Kommunikationspartner aus. Weiterhin ist zu beachten, dass der Agent selbst fehlerhafte Entscheidungen treffen kann. So muss neben einer Bestätigung des Nutzers auch das Einlegen eines Widerspruchs berücksichtigt werden. Zudem ist nicht auszuschließen, dass auch Handlungsziele und damit verbundene gestellte Aufgaben des Kommunikationsobjekts einer zeitlichen Veränderung unterliegen. Zuvor geäußerte Anfragen und Reaktionen können während des Dialogablaufs weiter konkretisiert oder auch korrigiert werden.

Ergebnis des Interpretationsvorgangs

Die unter Berücksichtigung des gegebenen Äußerungskontextes verwendeten *semantischen Grammatiken* liefern zu einer akzeptierten Zeichenfolge die assoziierte sprachunabhängige Bedeutungsstruktur. Diese muss nur die Menge *semantischer Anker* beinhalten, die auch tatsächlich informationstechnisch verarbeitet werden können. Beim Interpretationsprozess kann u. U. bereits eine einzelne *mikrolokale Grammatik* verschiedene Hypothesen zu dem perzipierten sprachlichen Verhalten liefern. Den ermittelten semantischen Einheiten kann allerdings anhand der Bewertung des Spracherkenners und der stochastischen *semantischen Grammatiken* ein Wahrscheinlichkeits- oder Konfidenzmaß zugeordnet werden. Wenn ein gewisses Maß an Unsicherheit bezüglich des Interpretationsergebnisses besteht, sind in diesem Fall konkrete Nachfragen an den Benutzer zu stellen. Eine Überprüfung der erkannten möglichen Absicht kann explizit erfolgen oder implizit durch die darauffolgenden Aktion durchgeführt werden. Wird keine Entscheidung getroffen und liegen somit mehrere *Merkmal-Werte-Relationen* als Interpretationsergebnis vor, so müssen diese zunächst parallel verwaltet werden. Zusätzlich können Ambiguitäten auch aus Datenbankabfragen resultieren. Auch in diesem Fall benötigt die Verhaltenssteuerung weitere Informationen zum Erreichen des Kommunikationsziels.

Eine erkannte elementare *Merkmal-Werte-Relation* stellt einen Bedeutungsträger zu einem Sprachzeichen dar. Die Bedeutung des Zeichengebrauchs in einer Situation ist

nach Meinung des Semiotikers CHARLES W. MORRIS nicht in einer Handlung des Interpretieren zu suchen, sondern in dem durch das Zeichen ausgelösten vorbereitenden Reiz. Diese *Disposition* kann zusammen mit anderen Stimuli Einfluss auf eine beobachtbare Reaktion haben. Allerdings muss durch perzipierte Zeichen nicht zwingend eine Handlung des Interpretieren hervorgerufen werden [Hör77, Nöt00]. Eine Bedeutung M (sofern für technische Systeme möglich) erschließt sich demnach erst durch die Ermittlung eines geschätzten Folgezustands \tilde{z}' der Umwelt und ggf. der Integration eines erkannten Bedeutungsträgers in den bestehenden Kontext. Unter diesen Annahmen liefert die pragmatische Informationsverarbeitung in einem technischen System zu der elementaren *Merkmal-Werte-Relation* weitere *semantische Anker* $\{c_i\} \in C$, die zu Ermittlung der Bedeutung M notwendig sind.

Referenzielle semantische Anker

Merkmal-Werte-Relationen dienen in einem technischen System u. a. zur Identifikation von auszuführenden Aktionen und der dafür notwendigen Parameter. Für diesen Zweck werden Bedeutungsstrukturen benötigt, die einen direkten Bezug zur modellierten Realität (z. B. Datenmodelle und Datenbanken) besitzen. Die durch den Interpretationsschritt generierten Bedeutungsträger können noch *semantische Anker* beinhalten, welche durch pragmatisches Wissen zu konkretisieren sind. Diese besitzen somit eine „Stellvertreterfunktion“. Ob und in welchem Ausmaß solche Referenzbeziehungen relevant sind, ist abhängig von der Funktionalität und der Domäne der Anwendung.

Allgemein kann man zwischen Referenzen auf Inhalte des bisherigen Dialogverlaufs (anaphorischen Ausdrücke) und *referenziellen semantischen Ankern* unterscheiden, welche durch Weltwissen ersetzt werden können.

Bei der Zuordnung von ermittelten elementaren MWR zu dem bestehenden Informationsstatus sind ggf. zuvor eingeführte Referenten zu berücksichtigen. Solche anaphorischen Bezüge müssen aber nicht zwingend in der elementaren Semantikrepräsentation direkt dargestellt werden. Dies ist z. B. in den Fällen möglich, in denen der Bezug auf den Referenten durch eine vorherige Frage des Agenten gegeben ist. Exemplarisch kann hierfür der Dialogschritt

Agent: *In welcher Stadt wohnt Peter?*

Benutzer: *Er wohnt in Cottbus.*

angeführt werden. In diesem Fall ist es möglich, die pronominale Referenz zu der Person „Peter“ bereits vorab herzustellen. Die Unifikation des *semantischen Ankers* (c_1 : „Cottbus“) zum Städtenamen mit einer bestehenden MWR ist hierbei eindeutig. Anders ist z. B. das sprachliche Verhalten „*Verwende dieselben Einstellungen wie beim letzten Mal.*“ zu interpretieren. Ein zugeordneter *referenzieller semantischer Anker* bezieht sich hier auf eine Menge von benutzerspezifischen Parametern zu einer ausführbaren Systemaktion, für die idealerweise eine bestehende *Merkmal-Werte-Relation* existiert.

Weiterhin muss für eine konkrete Datenbankabfrage auch die aktuelle Sprechsituation berücksichtigt werden. Dabei sind z. B. räumliche oder zeitliche Bezüge (Lokal- und Temporaldeixis) zur physikalischen Welt nötig. Sofern raumzeitliche Situationen nicht durch *semantische Grammatiken* adaptiv abgebildet werden, müssen diese Angaben zunächst durch *referenzielle semantische Anker* repräsentiert werden. Mögliche indexikalische Sprachzeichen sind beispielsweise „*nächste Woche Mittwoch*“, „*Ostersonntag*“ oder „*von hier*“. Einige tempusabhängige Bedeutungseinheiten lassen sich beispielsweise

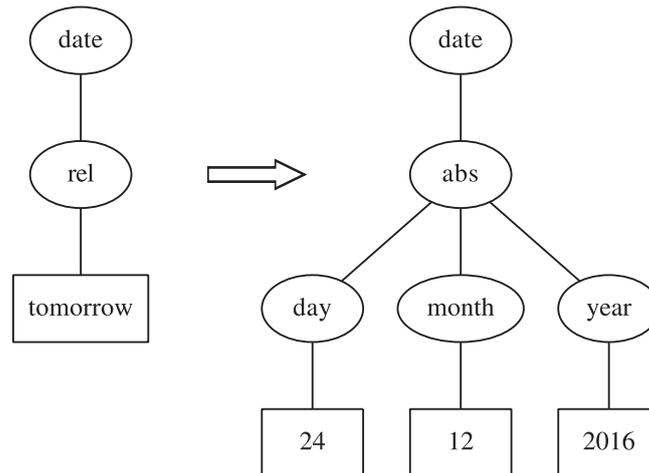


Abb. 4.4: Zusammenhang zwischen dem referenziellen semantischen Anker (links) und einem möglichen zugeordneten semantischen Anker (rechts)

mit Hilfe der Systemzeit bestimmen. So kann beispielsweise das zum Sprachzeichen „*morgen*“ assoziierte *Konzept* durch einen *semantischen Anker* substituiert werden. Dieser Zusammenhang ist Abb. 4.4 dargestellt.

4.3 Interpretation mit endlichen Äußerungs-Bedeutungs-Transduktoren

Äußerungs-Bedeutungs-Transduktoren T_{UM} haben in unserem Systemmodell die Aufgabe, das Spracherkennungsergebnis (endlicher Automat T_W) in einen semantischen Hypothesengraphen zu übersetzen (vgl. Abb. 4.5). Ein Worthypothesengraph T_W ist dabei so multisequenziell gestaltet, dass mögliche Wortfolgen jeweils einem durchgehenden Pfad π im Transduktor entsprechen [Wir12]. Das Ziel des semantischen Interpretationsprozesses ist die Erzeugung von *Merkmal-Werte-Relationen* (MWR) zu akzeptierten Sprachzeichenfolgen. Wie im Kapitel 4.2 beschrieben, ist eine *mikrolokale Grammatik* dabei so zu konzipieren, dass immer der gesamten Äußerung eine konkrete Bedeutungsrepräsentation zugeordnet wird. Diese Beziehung wird als *Äußerungs-Bedeutungs-Paar* (*utterance meaning pair*) [WHK⁺12, WL13] bezeichnet.

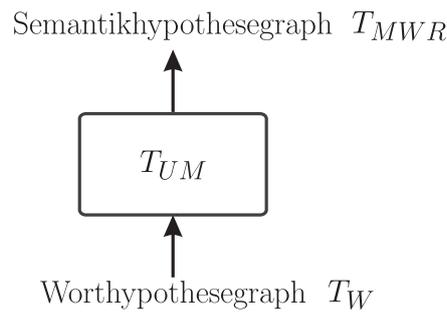


Abb. 4.5: Äußerungs-Bedeutungs-Transduktoren zur Interpretation von sprachlichen Realisierungen

Die Zuordnung einer elementaren *Merkmal-Werte-Relation* zu den im Transduktor T_W enthaltenen Sprachzeichenfolgen erfolgt durch die Komposition mit dem *Äußerungs-Bedeutungs-Transduktor* T_{UM} . Dies wird durch die Gleichung

$$T_{MWR} = T_W \circ T_{UM} \quad (4.4)$$

ausgedrückt. Für die Generierung einer Bedeutungsrepräsentation ist es notwendig, dass der *Äußerungs-Bedeutungs-Transduktor* T_{UM} zu einem gegebenen Kontext (*antecedents* A) mindestens eine von T_W erzeugte Sprachzeichensequenz akzeptiert. Eine direkte Übersetzung einer erkannten Sprachzeichensequenz (*beschriftete Totalordnung*) in eine *beschriftete partielle Ordnung* (vgl. Kapitel 4.1) ist mit *endlichen Transduktoren* allerdings nicht möglich. Aus diesem Grund wird die *Merkmal-Werte-Relation* zunächst einmal durch eine *MWR-Zeichenkette* repräsentiert. Diese Methode wurde bereits in [WW14] zur Sprachsteuerung eines beweglichen Mikrofonfelds vorgeschlagen und untersucht. Zu einer endlichen Menge von durchgehenden Pfaden $\mathcal{U}(I, F)$ im Kompositionsergebnis T_{MWR} erhält man eine entsprechende Anzahl an *MWR-Zeichenketten*. Dabei wird einem durchgehenden Pfad genau eine *MWR-Zeichenfolge* zugeordnet. Zu jeweils einer durch den Transduktor T_{MWR} generierten Symbolfolge kann anschließend in einem Nachbearbeitungsschritt ein *partielles Wort* erzeugt werden.

Zusammenfassend kann festgehalten werden, dass ein endlicher *Äußerungs-Bedeutungs-Transduktor* T_{UM} zu einer *mikrolokalen Grammatik* hier zwei wesentliche Funktionen erfüllen muss:

1. Akzeptor für mögliche Sprachäußerungen zum *semantischen Schema* S
2. Generator von *MWR-Zeichenketten*. Diese Symbolfolgen enthalten alle notwendigen Informationen zu Generierung einer *Merkmal-Werte-Relation*.

Die hier verwendeten *Äußerungs-Bedeutungs-Transduktoren* erzeugen direkt normalisierte Datenwerte.

Äußerungs-Bedeutungs-Transduktor als Wissensrepräsentation

Erstellte semantische Grammatiken werden situationsabhängig zur Interpretation von sprachlichen Zeichen eingesetzt. Die dazugehörigen *Äußerungs-Bedeutungs-Transduktoren* ordnen dem sprachlichen Verhalten eine *Merkmal-Werte-Relation* zu. Sie bilden ein Übersetzungsnetzwerk und beschreiben mögliche Beziehungen zwischen Zeichen. Durch die Automatenstruktur ist die chronologische Abfolge von Symbolen (*Syntax- bzw. Partiturmodell*) festgelegt. Zudem beschreibt die Klammerstruktur der Ausgabezeichenkette \mathbf{y} die Hierarchie der Bedeutungsrepräsentation. Anstatt eine Folge von einzelnen semantischen *Konzepten* wird hier eine Zeichenkette zur *Merkmal-Werte-Relation* generiert.

Parallele Wege eines *Äußerungs-Bedeutungs-Transduktors* mit denselben Ein- oder Ausgabezeichen beschreiben *paradigmatische Relationen*. So zum Beispiel können *Synonyme* oder *synonymischen Syntagmen* angegeben werden. In diesem Fall besteht ein Zusammenhang zwischen der Menge der Wege $\mathcal{U}(z_i, \infty, \mathbf{y}, z_j)$, mit $z_i, z_j \in Z; \mathbf{y} \in Y^*$ und beliebiger Eingabezeichenfolge ∞ , welche dieselbe Ausgabesymbolfolge generieren.

Semantische Strukturen basieren auf *domänenspezifischem Wissen*. Wurden bei der Modellierung des Transduktors T_{UM} Platzhalter verwendet, so beschreiben diese Beziehungen zwischen austauschbaren semantischen Einheiten in einem definierten Kontext. Dies können beispielsweise alle zu berücksichtigenden Werte eines Attributs sein. Quellen möglicher Datenwerte sind u. a. spezielle Lexika oder Datenbankeinträge.

MWR-Zeichenketten

Die Syntax der durch Äußerungs-Bedeutungs-Transduktor T_{UM} generierten *MWR-Zeichenketten* lässt sich zum Beispiel durch eine kontextfreie Grammatik oder alternativ durch die folgende vereinfachte erweiterte Backus-Naur-Form (eBNF) [WW14]:

```

FVR      = "FVR" state
state    = "[" states statedef states "]"
states   = [ [ states ] state ]
statedef = label [ ":" weight ]
label    = string
weight   = numeric | "NaN"

```

ausdrücken. Zu der Menge der Terminalsymbole X gehören die Zeichen, welche in Anführungszeichen gesetzt sind, wie z. B. "FVR" und "[". Das Startsymbol $S \in N$ ist in diesem Fall FVR. Eine Option wird in der eBNF durch die eckigen Klammern ohne Anführungszeichen [...] ausgedrückt. Die Nichtterminale *string* und *numeric* sind in der angegebenen eBNF nicht weiter durch Produktionsregeln beschrieben. Für das Zeichen *string* können mit einer optionalen Wiederholung Buchstaben, Ziffern und Sonderzeichen eingesetzt werden. Diese Symbole werden zur Beschriftung der erzeugten Halbordnung verwendet. Sie sind somit als Elemente der endlichen Etikettenmenge $L \subset X$ der MWR zu betrachten. Die Darstellung des berechneten Gewichts eines Zu-

standsübergangs erfolgt durch das Nichtterminal *numeric*, wobei hier in der Regel als Dezimaltrennzeichen ein Punkt verwendet wird. Liegen keine Gewichte vor, so wird dies in der *Merkmal-Werte-Relation* durch das Wort „NaN“ (not a number) markiert. Die Ableitung der verwendeten Klammergrammatik kann als gewurzelter Baum dargestellt werden. Dieser Ableitungsbaum entspricht der Darstellung einer *Merkmal-Werte-Relation*. Zur Veranschaulichung der Gestalt solcher *MWR-Zeichenfolgen* dient das bereits in [Lin14] vorgestellte, nachfolgend beschriebene Szenario.

Beispiel Bahnauskunftssystem

Ausgangspunkt ist jeweils eine erkannte Sprachäußerung der deutschen Laut- und Gebärdensprache. Diese sollen aus dem Bereich der Bahnauskunftssysteme stammen. Der Zusammenhang zwischen den Sprachäußerungen sowie der ungewichteten *MWR-Zeichenfolgen* ergibt sich dabei folgendermaßen:

Wortfolge der deutschen Lautsprache:

Ich möchte morgen von Berlin nach Dresden fahren.

Glossennotation für gleichbedeutende Realisierung in DGS:

*MORGEN INDEX-ich WUNSCH BERLIN INDEX-li DRESDEN
INDEX-re li-FAHREN-re*

mögliche ungewichtete *MWR-Zeichenfolge*:

FVR[[date_rel[tomorrow]][loc[Berlin]][loc[Dresden]]route_abs[direction[1to2]]]

nachbearbeitete *MWR-Zeichenfolge*:

FVR[route_abs[date_rel[tomorrow]][loc[Berlin]][loc[Dresden]]][direction[1to2]]]

In diesem Beispiel ist das *semantische Schema*, zu dem die beiden aufgeführten Sprachäußerungen zugeordnet wurden, mit *route_abs* benannt. In derselben Sprechsituation ist der lautsprachliche Satz *“Von Berlin nach Dresden möchte ich morgen fahren.”* gleichbedeutend zu der bereits oben aufgeführten Wortfolge *“Ich möchte morgen von Berlin nach Dresden fahren.”* anzusehen.

Die Reihenfolgeinformationen aus der Syntax der verwendeten Sprachzeichen sind bei partiellen Wörtern nicht mehr vorhanden. Auf der Bedeutungsebene sind *semantische Anker* meist simultan zu betrachten.

Jedoch benötigt man in bestimmten Fällen die Reihenfolgeinformation semantischer Einheiten. In dem aufgeführten Beispiel ist es erforderlich, einen genauen Bezug zwischen den Ortsangaben und den Parametern der zu berechnenden Route herzustellen. Die Phrasen *“von Berlin“* und *“nach Dresden“* definieren den Start- und Zielpunkt. Bestehende syntaktische Beziehungen zwischen Wörtern besitzen demnach auch eine

bedeutungsunterscheidende Funktion.

Für die Gebärdensprache ist diese Information durch die Bewegungsrichtung des *Raumverbs* FAHREN gegeben. Hierzu wurden die Städtebezeichnungen zuvor im Gebärdensraum verortet. Prinzipiell ist eine zeitliche Abfolge zugeordneter *semantischer Anker* für dieses Beispiel nicht relevant. Die Modellierung wurde hier aber so gewählt, dass sich die nötige Information durch die Reihenfolge der Ortsangaben in der *MWR-Zeichenkette* ergibt. Alternativ kann die durch *“direction“* dargestellte semantische Information durch unterschiedliche Attribute für Start- und Zielort modelliert werden.

Die beschriebene Problematik wird jedoch akut, wenn man komplexere Eingaben betrachtet. So kann man sich beispielsweise auch eine Routenanfrage vorstellen, welche den Startpunkt A, die Zwischenziele B und C und den Zielort D beinhaltet. Hier führt eine unterschiedliche Reihenfolge einer Aufzählung von Zwischenaufhalten zu verschiedenen Ergebnissen. Dies muss bei der Generierung von *Merkmals-Werte-Relationen* berücksichtigt werden. Eine Möglichkeit, diese Abfolgeinformationen in der MWR abzubilden, wurde in [WW14] vorgestellt. Dabei erfolgt eine Kaskadierung hierarchisch gleichrangiger semantischer Einheiten mit derselben Beschriftung. Endliche Transduktoren sind allerdings nicht in der Lage, die erzeugte *MWR-Zeichenkette* so zu verändern, dass durch die Syntax der Klammerstruktur die Kaskadierung gleichnamiger Geschwisterknoten direkt erreicht werden kann. Eine direkte Weiterverarbeitung des Interpretationsergebnisses mit endlichen Automaten ist deshalb nicht möglich. Zu diesem Zweck wurde am Lehrstuhl ein Parser implementiert, welcher die generierte *MWR-Zeichenfolge* syntaktisch analysiert und eine *beschriftete Halbordnung* erzeugt. Das auf diese Weise erstellte MWR-Objekt kann als Baumstruktur visualisiert werden. In dieser bildet die Bezeichnung des detektierten *semantischen Schemas* sinnvollerweise die Wurzel des Baumes. Zu der oben angegebenen Beispieläußerung ist die aus der *MWR-Zeichenkette* erzeugte Baumstruktur durch Abb. 4.6 gegeben.

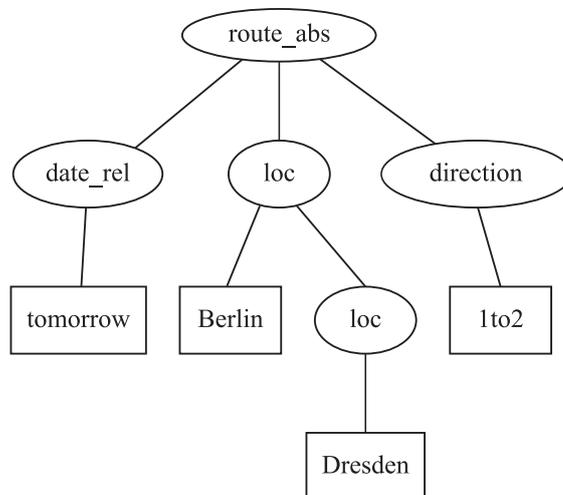


Abb. 4.6: MWR als Baumstruktur zum Beispiel Bahnauskunftssystem [Lin14]

4.4 Artikulation mit endlichen Äußerungs-Bedeutungs-Transduktoren

Ausgangspunkt der nachfolgenden Betrachtung ist das Vorliegen einer konkreten *Merkmal-Werte-Relation* (MWR). Das Ziel der Artikulation besteht in der Generierung von Sprachzeichenfolgen, welche zu der gegebenen *Merkmal-Werte-Relation* passen (vgl. Kapitel 3.2). An dieser Stelle wird darauf hingewiesen, dass die sprachenunabhängige Inhaltsfestlegung ein wichtiger Teil der Textgenerierung ist. Bei den Ausführungen in dieser Arbeit wird jedoch auf die Makroplanung nicht näher eingegangen und die interpretierte Bedeutungsstruktur direkt für die Artikulation verwendet.

Für die Sprachgenerierung sollen analog zur Perzeptionsseite *Bedeutungs-Äußerungs-Transduktoren* verwendet werden. Die Generierung von Sprachäußerungen aus einer gegebenen Informationsstruktur erfolgt durch die folgenden 3 Schritte:

1. Linearisierung der *Merkmal-Werte-Relation*. Dazu wird ein endlicher Transduktor T_{MWP} erstellt, der alle möglichen Folgen von *Merkmal-Werte-Permutationen* zu der MWR generiert.
2. Zuordnung möglicher Sprachzeichenfolgen zu den erzeugten Symbolfolgen einer *Merkmal-Werte-Relation*. Die Funktion des Akzeptors und Generators übernehmen in diesem Fall *Bedeutungs-Äußerungs-Transduktoren*.
3. Auswahl einer generierten Sprachäußerung. Optional erfolgt zuvor eine Bewertung des erzeugten Wortgraphen mittels eines Sprachmodells der Zielsprache.

Transduktor T_{MWP} zur Generierung von MWR-Zeichenketten

Wenn die semantische Verarbeitung durch endliche Automaten erfolgt, dann kann eine gegebene *Merkmal-Werte-Relation* nicht direkt in eine Wortfolge übersetzt werden. Es ist in diesem Fall eine *Linearisierung* der Bedeutungsstruktur notwendig. Dabei werden zunächst zu einem vorgegebenen beschrifteten *partiellen Wort* alle möglichen *Merkmal-Werte-Permutationen* generiert. Eine *MWR-Zeichenkette* stellt genau dann eine erlaubte *Merkmal-Werte-Permutation* der Bedeutungsstruktur dar, wenn durch das Parsen dieser Klammerstruktur dieselbe *Merkmal-Werte-Relation* erzeugt wird. Mit diesen Variationen von *MWR-Zeichenketten* wird der endliche Automat T_{MWP} erzeugt, welcher multisequenziell alle möglichen *Merkmal-Werte-Permutationen* generiert. Die Ausgabesymbolfolgen der durchgehenden Pfade $\mathcal{U}(I, F)$ des Transduktors T_{MWP} müssen dabei der Syntax der verwendeten *MWR-Zeichenketten* entsprechen.

Zur Generierung aller möglichen Folgen solcher Sequenzen erfolgt eine Untersuchung der Baumstruktur. Ausgangspunkt ist die Wurzel des Baumes, welche den Startknoten für eine *Tiefensuche* bildet. Dabei wird der gerichtete Graph rekursiv exploriert, wobei

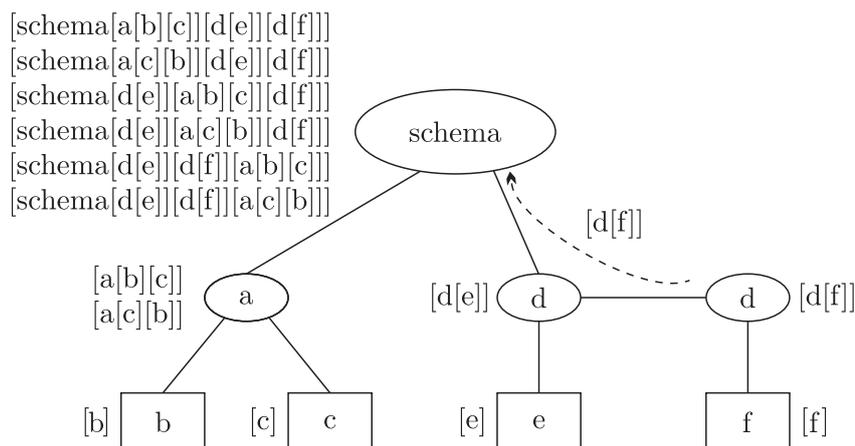


Abb. 4.7: Visualisierung der Konstruktion möglicher Permutationen

zunächst immer der erste Nachfolgeknoten untersucht wird. Diese Expandierung erfolgt so lange in die Tiefe, bis ein Blatt der Bedeutungsstruktur detektiert worden ist. Dessen Beschriftung entspricht in der Regel einem Datenwert. Beim Rückwärtsgehen werden die Etiketten der Knoten eingesammelt. Wenn von einem Knoten alle Kinder abschließend untersucht wurden, so übergibt dieser seine Ergebnisliste an den Elternknoten. Diese Liste enthält alle Permutationen von Beschriftungsfolgen zu den untergeordneten Knoten.

Eine Besonderheit besteht bei gleichnamigen Eltern-Kind-Knoten. Diese entstehen durch Kaskadierung gleicher Attributbezeichnungen bei der Erstellung einer *Merkmal-Werte-Relation*. In diesem Fall ist die \leftarrow -Relation dieser Kindknoten unter Beibehaltung der Reihenfolge aufzuheben. Die aufgesammelten Beschriftungen zu dem gleichnamigen untergeordneten Knoten werden in einer separaten Liste geführt. Bei der Kaskadierung von mehr als zwei Werten wird dieser Prozess entsprechend wiederholt durchgeführt, bis keine gleichnamigen Elternknoten auftreten. Erst auf dieser Hierarchieebene erfolgt dann die Permutation dieser Zeichenfolgen. Dieser Vorgang ist in Abb. 4.7 durch die gestrichelte Linie angedeutet. Hierbei werden die Folgen nicht in die Ergebnisliste aufgenommen, bei der die ursprüngliche Reihenfolge kaskadierter Pfade nicht gegeben ist.

Als weiteres Beispiel dient die bereits in der Abb. 4.6 abgebildete Bedeutungsrepräsentation aus dem Beispiel „Bahnauskunftssystem“. Eine modifizierte Darstellung dieser *Merkmal-Werte-Relation*, in der die Elemente hierarchisch angeordnet sind, ist durch Abb. 4.8 gegeben. Die Städtenamen sind beide der semantischen Kategorie „loc“ zugeordnet und somit hierarchisch gleichrangig zu betrachten. Für dieses Beispiel ergeben sich mit den 4 gleichrangigen Attributen $4! = 24$ Kombinationen. Dabei bleiben die Folgen unberücksichtigt, in denen das *Merkmal-Wert-Paar* $[loc[Dresden]]$ vor der Zeichenfolge $[loc[Berlin]]$ auftritt.

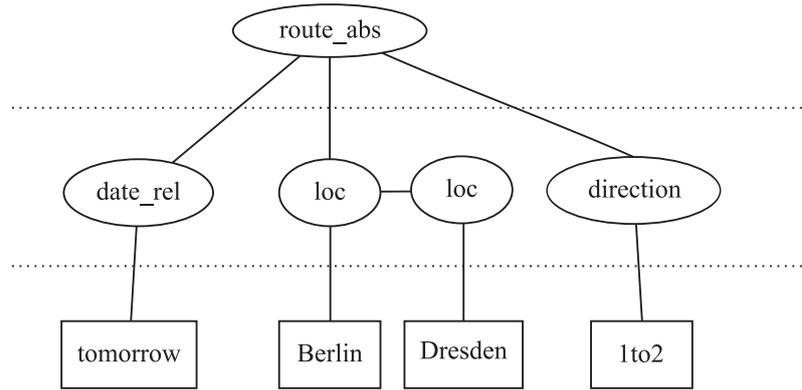


Abb. 4.8: Hierarchieebene innerhalb einer MWR

Somit erhält man die folgenden $4! : 2! = 12$ möglichen *MWR-Zeichenketten*:

1. $FVR[route_abs[date_rel[tomorrow]][loc[Berlin]][loc[Dresden]][direction[1to2]]]$
2. $FVR[route_abs[loc[Berlin]][date_rel[tomorrow]][loc[Dresden]][direction[1to2]]]$
3. $FVR[route_abs[loc[Berlin]][loc[Dresden]][date_rel[tomorrow]][direction[1to2]]]$
4. $FVR[route_abs[direction[1to2]][date_rel[tomorrow]][loc[Berlin]][loc[Dresden]]]$
5. $FVR[route_abs[date_rel[tomorrow]][direction[1to2]][loc[Berlin]][loc[Dresden]]]$
6. $FVR[route_abs[loc[Berlin]][direction[1to2]][date_rel[tomorrow]][loc[Dresden]]]$
7. $FVR[route_abs[direction[1to2]][loc[Berlin]][date_rel[tomorrow]][loc[Dresden]]]$
8. $FVR[route_abs[date_rel[tomorrow]][loc[Berlin]][direction[1to2]][loc[Dresden]]]$
9. $FVR[route_abs[loc[Berlin]][date_rel[tomorrow]][direction[1to2]][loc[Dresden]]]$
10. $FVR[route_abs[loc[Berlin]][loc[Dresden]][direction[1to2]][date_rel[tomorrow]]]$
11. $FVR[route_abs[direction[1to2]][loc[Berlin]][loc[Dresden]][date_rel[tomorrow]]]$
12. $FVR[route_abs[loc[Berlin]][direction[1to2]][loc[Dresden]][date_rel[tomorrow]]]$

In einem nächsten Verarbeitungsschritt ist aus diesen Folgen der Transduktor T_{MWP} zu erzeugen. Dazu sind die *MWR-Zeichenketten* anhand der Klammerstruktur so zu segmentieren, dass die separierten Wörter Elemente des Ausgabealphabets Y der *Äußerungs-Bedeutungs-Transduktoren* T_{UM} bilden. Die erzeugte Liste mit den segmentierten Sequenzen von *Merkmal-Werte-Permutationen* kann dann entsprechend ausgelesen werden. Eine Zählvariable über alle in der Liste enthaltenen Wörter sorgt für die korrekte Zuordnung zu den Zuständen aus dem Zustandsalphabet Z des Transduktors T_{MWP} . Zustände am Ende einer jeden *MWR-Zeichenfolge* bilden die Menge der Schlusszustände $F \subseteq Z$.

Bedeutungs-Äußerungs-Transduktoren

Während *Äußerungs-Bedeutungs-Transduktoren* T_{UM} zu Wortfolgen mögliche *MWR-Zeichenketten* zuordnen, ist die Aufgabe der *Bedeutungs-Äußerungs-Transduktoren* umgekehrt. Diese dienen als

- Akzeptor für *MWR-Zeichenketten* und als
- Generator von Symbolfolgen der Zielsprache.

Folglich ist es die einfachste Möglichkeit, die modellierten *Äußerungs-Bedeutungs-Transduktoren* T_{UM} auch für die Artikulation zu verwenden. Die Automatenoperation *Inversion* (vgl. Kapitel 2.4.2), bei der Ein- und Ausgabezeichen an den Zustandsübergängen vertauscht werden, liefert das gewünschte Ergebnis. Somit gilt für die *Bedeutungs-Äußerungs-Transduktoren* der Zusammenhang:

$$T_{MU} = T_{UM}^{-1} \quad (4.5)$$

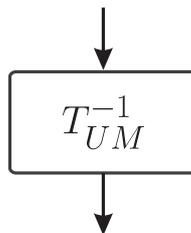
Für die *Bedeutungs-Äußerungs-Transduktoren* wird in der Arbeit die Abkürzung $T_{MU} = T_{UM}^{-1}$ verwendet. Dieser inverse Transduktor enthält die nötigen syntaktischen und semantischen Informationen. Er generiert zu akzeptierten *MWR-Zeichenketten* die Menge von Sprachäußerungen zur gegebenen Bedeutung M .

Durch die Komposition des *Bedeutungs-Äußerungs-Transduktors* mit dem Transduktor T_{MWP} wird ein Automat T_{AW} erzeugt, welcher die artikulierten Wortfolgen beinhaltet.

$$T_{AW} = T_{MWP} \circ T_{UM}^{-1} \quad (4.6)$$

Hier entspricht das Ausgabealphabet Y des Transduktors T_{MWP} dem Eingabealphabet des *inversen Äußerungs-Bedeutungs-Transduktors* T_{UM}^{-1} . In einem Sprachdialogsystem werden für die Interpretation und Artikulation trotz einzelner gemeinsamer *semantischer Anker* oft unterschiedliche Bedeutungsstrukturen verwendet. In diesem Fall müssen die *Bedeutungs-Äußerungs-Transduktoren* separat modelliert werden.

Merkmal-Werte-Permutationen T_{MWP}



Artikulierter Worthypothesegraph T_{AW}

Abb. 4.9: Äußerungs-Bedeutungs-Transduktoren zur Artikulation von Sprachzeichenfolgen

Durchführung der Artikulation

Nach Gleichung 4.6 erhält man durch Komposition des generierten Transduktors T_{MWP} mit dem *inversen Äußerungs-Bedeutungs-Transduktor* T_{UM}^{-1} den Automaten T_{AW} (vgl. Abb. 4.9). Dieser generiert alle möglichen Sprachzeichenfolgen zu den akzeptierten Sequenzen von *Merkmals-Werte-Permutationen*. In Abhängigkeit von der Modellierung des *Äußerungs-Bedeutungs-Transduktors* kann der Generierungsprozess auch Sprachzeichenfolgen hervorbringen, die nicht der Grammatik der natürlichen Zielsprache entsprechen.

Definition 4.1. *Eine Sprachzeichenfolge der natürlichen Sprache \mathcal{L} wird als **grammatisch stimmig** oder **wohlgeformt** bezeichnet, wenn diese nach Ansicht der Sprecher von \mathcal{L} ein grammatisch korrekter Ausdruck ist.*

Für den Fall dass eine generative Grammatik nicht wohlgeformte Wortfolgen der Zielsprache erzeugt, können z. B. die bereits im Kapitel 2.4 erwähnten statistischen Sprachmodelle zum Einsatz kommen. Diese enthalten A-priori-Kenntnisse zu der jeweiligen Zielsprache. In diesem Fall kann der Artikulationsprozess durch

$$T_{AW} = T_{MWP} \circ T_{UM}^{-1} \circ T_{SM}. \quad (4.7)$$

ausgedrückt werden. Typischerweise erfolgt jedoch bei der optionalen Anwendung eines *N-Gramm-Sprachmodells* keine Ablehnung von grammatisch unstimmen Sprachzeichensequenzen. Allerdings werden die Wortfolgen anhand ihrer geschätzten Auftretenswahrscheinlichkeit in der Sprache bewertet. Im Kapitel 6.3 wird die Verwendung von Sprachmodellen für die Bewertung von generierten Wortfolgen der deutschen Lautsprache thematisiert.

4.5 Zum Einsatz von Petrinetztransduktoren in der semantischen Sprachverarbeitung

Wie in Kapitel 4.1 dargelegt, sind *beschriftete partielle Ordnungen* geeignet um semantische Daten zu repräsentieren. Solche Halbordnungen können z. B. auch das nebenläufige Prozessverhalten in verteilten Systemen dokumentieren. *Petrinetze* stellen ein formales Modell zur Beschreibung dieser nebenläufigen Ereignisfolgen dar. Die Idee, *Petrinetz-Transduktoren* für die Entwicklung intelligenter Maschinen einzusetzen, gibt es schon seit längerer Zeit [Bil88, WS89, WKTS91, LS92]. Prinzipiell eignen sich diese Automaten auch für die semantische Verarbeitung in sprachverarbeitenden Systemen [HKLW08]. In den letzten Jahren wurden *gewichtete Petrinetz-Transduktoren* algebraisch beschrieben und notwendige Automatenoperationen entwickelt.

Definition 4.2. Petrinetz-Transduktoren (PNT) sind Automaten, welche die Übersetzung zwischen partiellen Sprachen realisieren. Ein **gewichteter** PNT über einen Bisemiring $\mathbb{S} = (\mathbb{K}, \oplus, \otimes, \boxtimes, \bar{0}, \bar{1})$ ist algebraisch betrachtet ein Tupel $N = (P, T, F, W, p_I, p_F, X, \sigma, Y, \delta, \omega)$ mit

- der endlichen Menge von Stellen (Plätzen) P ,
- einer endlichen Menge von Transitionen T , mit $T \cap P = \emptyset$,
- der endlichen Kantenmenge (Flussrelation) $F \subseteq (P \times T) \cup (T \times P)$,
- der Kantengewichtungsfunktion $W : F \rightarrow \mathbb{N}_0$,
- einer initialen Markierung (Anfangsmarkierung) $m_0 = p_I$, mit $p_I \in P$,
- einer finalen Markierung $m_F = p_F$, mit $p_F \in P$,
- dem endlichen Eingabealphabet X ,
- der Eingabefunktion $\sigma : T \rightarrow (X \cup \{\epsilon\})$
- dem endlichen Ausgabealphabet Y ,
- der Ausgabefunktion $\delta : T \rightarrow (Y \cup \{\epsilon\})$ und
- der Verhaltensfunktion $\omega : T \rightarrow \mathbb{K}$ [LHW14].

Petrinetze besitzen zwei verschiedene Knotentypen. Dies sind Stellen (als Kreise dargestellt) und Transitionen (Rechtecke des Graphen). Eine Markierung m ordnet jeder Stelle $p \in P$ den Wert $m(p) \in \mathbb{N}_0$ zu. Gebräuchlich ist die Verwendung eines schwarzen Punktes zur Kennzeichnung der Token. Diese sind bei den Stellen aufgeführt, welche im aktuellen Zustand entsprechende Marken besitzen. Das Schalten bzw. Feuern einer Transition $t_i \in T$ in einer Markierung m ist nur möglich, wenn für die betrachtete Transition t_i alle Vorgängerstellen mit einer Mindestanzahl von Token belegt sind. Die Anzahl der für das Schalten erforderlichen Token ergibt sich aus dem Kantengewicht W (Flusszahl). Zusätzlich müssen die Kapazitäten aller Nachfolgestellen von t_i für die Aufnahme von Token ausreichend sein.

Petrinetz-Transduktoren können als gerichtete, bipartite Graphen dargestellt werden. In der Abb. 4.10 ist der Graph eines einfachen sequenziellen PNT dargestellt. Die erste Transition ist in der dargestellten Markierung schaltbereit (aktiviert), da die Vorgängerstelle p_I eine Marke besitzt. Das Schalten einer Transition $m \xrightarrow{t} m'$ kann als Ereignis bzw. als ein dynamisches Systemverhalten betrachtet werden und führt zur Folgemarkierung m' . Dabei werden die Token in den Nachfolgestellen abgelegt.

Bei einem *gewichteten Petrinetz-Transduktor* ordnet die Verhaltensfunktion ω den Transitionen entsprechende Gewichte aus dem Bisemiring \mathbb{S} zu. Dieser Bisemiring ist ein 6-Tupel $\mathbb{S} = (\mathbb{K}, \oplus, \otimes, \boxtimes, \bar{0}, \bar{1})$ mit der Menge \mathbb{K} sowie den binären Operationen Addition \oplus , sequenzielle Multiplikation \otimes und parallele Multiplikation \boxtimes . Der Bisemiring \mathbb{S} besteht aus zwei Halbringen, dem Halbring $(\mathbb{K}, \oplus, \otimes, \bar{0}, \bar{1})$ und dem kommutativen Halbring

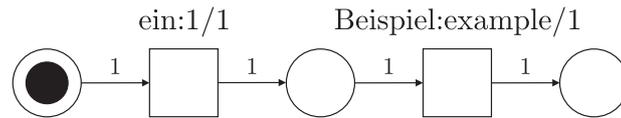


Abb. 4.10: Graph eines einfachen PNT mit der Anfangsmarkierung $p_I = m_0$

$(\mathbb{K}, \oplus, \boxtimes, \bar{0}, \bar{1})$ mit jeweils derselben Menge \mathbb{K} . Die Halbgruppe (\mathbb{K}, \boxtimes) ist hier kommutativ, besitzt jedoch kein neutrales Element [FKM09, LHW14]. Der Markierungswechsel in einem *Petrinetz-Transduktor* ist dabei vergleichbar mit dem Zustandsübergang in einem endlichen Automaten. Beim Schalten einer Transition liest ein PNT das Eingabewort x des Alphabets X und schreibt das Zeichen y des Ausgabealphabets Y . Eine aktivierte Transition muss nicht zwingend schalten. In einem durch die Stellenmarkierung des PNT gegebenen Systemzustand können hingegen auch mehrere Transitionen gleichzeitig schalten. Mit Petrinetzen können auf diese Weise dynamische nebenläufige Prozesse gut beschrieben werden. Wir haben gesehen, dass an den Transitionen eingelesene Sprachzeichen in Ausgabewörter übersetzt werden können. Auf diese Weise akzeptiert ein PNT eine Menge *partieller Ordnungen* der *partiellen Eingabesprache*. Zu einem akzeptierten partiellen Wort wird wiederum durch die Ausgabefunktion eine *etikettierte partielle Ordnung* generiert. Hierbei werden nur die Durchläufe betrachtet, welche im initialen Zustand p_I starten und in der finalen Stelle p_F münden.

Gewichtete Petrinetz-Transduktoren sind ein geeignetes Mittel für die Generierung und auch zur Darstellung von Bedeutungsstrukturen. Sie können als Verallgemeinerung der endlichen Transduktoren angesehen werden. Durch entsprechende Automatenoperationen, wie die Komposition von *Petrinetz-Transduktoren*, können diese möglicherweise auch für die komplette semantische Verarbeitung in einem kognitiven System verwendet werden. Auf diese Weise ist es möglich, semantische Strukturen zu vereinigen oder auch einzelne Teile miteinander zu verketteten. Dazu wurden wichtige Operationen für PNTs, wie Komposition, Kleenescher Abschluss, Summe und die Verkettung (Produkt) bereits in [LHW14] beschrieben. Für die Erzeugung *beschrifteter partieller Wörter* bei der Interpretation und auch für die Generierung multisequenzieller Folgen bei der Artikulation müssen derzeit Algorithmen verwendet werden, welche nicht durch *endliche Automaten* abgebildet werden können (vgl. Kapitel 4.3 und 4.4). Prinzipiell kann aus jedem FST ein äquivalenter PNT erzeugt werden [LH13], so dass die in dieser Arbeit gewonnenen Erkenntnisse zukünftig auch auf *Petrinetz-Transduktoren* angewandt werden können. Weiterführende Informationen zu den Themen Petrinetze und gewichtete Petrinetztransduktoren findet man u. a. in [Rei91, Mau06, Tab06, LH12, Lun12, LH13, LHW14].

Kapitel 5

Modellierung von Grammatiken für ein semantisches Schema

Vorhandene *Äußerungs-Bedeutungs-Transduktoren* sind Voraussetzung für die Untersuchung des in Kapitel 4 vorgestellten Konzepts zur semantischen Sprachverarbeitung. Diese werden in einer Beispielanwendung zur Interpretation und Artikulation von Sprachäußerungen der deutschen Laut- und der Deutschen Gebärdensprache eingesetzt. In diesem Abschnitt wird eine Möglichkeit der systematischen Modellierung von Transduktoren zu *mikrolokalen Grammatiken* beschrieben. Dafür wird jeweils eine *Elementargrammatik* benötigt. Durch Abstraktion wird der *Äußerungs-Bedeutungs-Transduktor* zu einem *semantischen Schema* erstellt.

5.1 Beschreibung der Bedeutungseinheiten zur Elementargrammatik

Ausgangspunkt der Konstruktion einer *Elementargrammatik* ist die Beschreibung einer konkreten Situation, in der eine endliche Menge von Sprachäußerungen $\mathcal{U}(M)$ zum Ausdruck einer elementaren *Merkmal-Werte-Relation* möglich sind.

Als Vertreter der zu erstellenden *Elementargrammatik* für die deutsche Lautsprache dient der Satz: „Dort drüben steht Peter und wartet.“ Dieser Aussagesatz beschreibt einen konkreten Aspekt, einer aktuell bestehenden Situation. Dieser ist in Abb. 5.1 schematisch dargestellt. Möglicherweise ist dieses verbale Verhalten eine Reaktion auf eine zuvor gestellte Frage. In diesem Äußerungskontext kann der zusammengesetzte Ausdruck sprachlich einzeln interpretiert werden. Mit Hilfe von linguistischem Wissen (sprachlicher Konventionen) können folgende semantischen Einheiten zu der

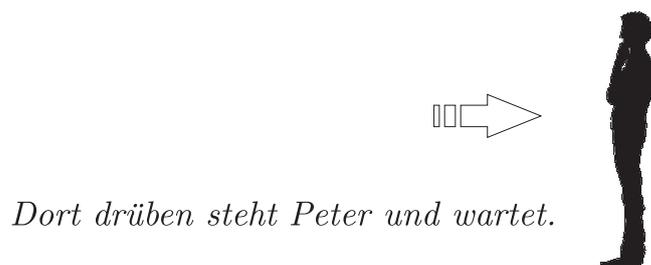


Abb. 5.1: Visualisierung der Beispielaussage

Sprachäußerung referenziert werden:

- Die **relative Ortsangabe** *dort drüben*.
Dieses Sprachzeichen drückt aus, dass sich ein Subjekt oder Objekt an einem räumlichen Punkt befindet, welcher nicht explizit benannt ist. Die tatsächliche Position ergibt sich möglicherweise aus dem Kontext der Sprachäußerung oder aus der vorherrschenden Gesprächssituation. So kann in diesem Fall auch eine Gestik des Sprechers zusätzliche Informationen liefern. Nachfolgend ist der zu der relativen Ortsangabe „dort drüben“ zugeordnete *semantische Anker* durch die Klammerstruktur $[loc[rel[there]]]$ angegeben.
- Die **Aktionen** *stehen* und *warten*.
Aktionen beschreiben Prozesse. Das Verb „stehen“ weist in diesem Fall darauf hin, dass sich jemand oder etwas an einem Platz aufhält. Anders als beim „sitzen“ oder „liegen“ befindet sich das Objekt oder Subjekt in einer aufrechten Position. Zudem hat die Aktion „stehen“ einen Ortsbezug. Die Tätigkeit des Wartens drückt in diesem Sprechakt aus, dass jemand oder etwas an einem Ort bzw. auch in seinem Zustand verharrt, bis ein bestimmtes Ereignis eintritt. Die verwendete Notation für die zugeordneten semantischen Einheiten ist hierfür $[action[stand]]$ bzw. $[action[wait]]$.
- Eine **Person**, welche in diesem Fall lediglich durch ihren Vornamen *Peter* bestimmt ist. Der vom Sprecher verwendete Name „Peter“ bezieht sich hier auf eine nicht näher bestimmte Person. Das zu dem Vornamen „Peter“ zugeordnete *Konzept* wird in dieser Arbeit durch $[person[name[first_name[Peter]]]]$ notiert.

Eine Interpretation der gesamten Sprachäußerung muss allerdings unter Berücksichtigung der Beziehungen der Zeichen untereinander und des bestehenden Äußerungskontextes erfolgen. Die Bedeutung des verbalen Verhaltens besteht hier in der Referenz zu einem Bezugsobjekt (Person), mit den angegebenen wirklichkeitsbezogenen Eigenschaften (Vorname der Person) und Zuständen (Position der Person und ausgeführte Handlungen). Die Festlegung einer sinnvollen Merkmalshierarchie und damit verbunden auch die Bezeichnung der Attribute sind stark vom Anwendungsgebiet abhängig.

Zur Veranschaulichung der Kaskadierung *semantischer Anker* in der *Merkmal-Werte-Relation* und deren Auswirkung bei der Artikulation erfolgt eine Berücksichtigung der Reihenfolge der Handlungen *stehen* und *warten*. Es wird idealisiert angenommen, dass die beiden bedeutungsähnlichen *Merkmal-Werte-Relationen* derselben Bedeutung *M* und somit auch derselben *Elementargrammatik* zugeordnet werden.

Analog zur deutschen Lautsprache wird auch für die Deutsche Gebärdensprache eine *Elementargrammatik* benötigt. Zu diesen Sprachäußerungen muss der Interpretationsprozess dieselben semantischen Einheiten zuordnen. Die Bezeichnungen der verwendeten *semantischen Konzepte* sollte bedacht gewählt werden. Um die Sprachunabhängigkeit der Bedeutungseinheiten zu betonen, werden in der Beispielanwendung hierfür Begriffe der englischen Sprache verwendet.

Namengebärden in der Deutschen Gebärdensprache

In der Gebärdensprachgemeinschaft werden für Personen, abweichend vom lautsprachlichen bzw. amtlichen Namenssystem (mit Vor- und Zuname), eigene Gebärdennamen verwendet [Uhl12]. Ist einem Gesprächspartner die Namengebärde einer Person unbekannt, so wird bei deren Einführung zusätzlich auch der lautsprachliche Name durch das Fingeralphabet dargestellt. Für die Ausführung der meist produktiven Namengebärden gibt es keine allgemeingültigen Konventionen. Teilweise beziehen sich die jeweiligen Namengebärden auf typische Eigenschaften (Verhalten, Charakter), Hobbys und Berufe, auffällige physische Merkmale oder andere typische Äußerlichkeiten der jeweiligen Personen. Solche beschreibende Namengebärden können auch einer zeitlichen Begrenzung unterliegen. Für den Fall dass die Namengebärde mit der Zeit unpassend geworden ist, kann einem Menschen auch ein neuer Name gegeben werden.

Namengebärden können allerdings auch mit dem lautsprachlichen bzw. amtlichen Namen in Zusammenhang gebracht werden. So erfolgt für die Kreation der Gebärde oft auch ein Bezug zum Fingeralphabet. Dabei wird häufig der 1. Buchstabe des amtlichen Namens für die Namengebärde verwendet. Gegebenenfalls kann auch eine Assoziation zum Familiennamen (z. B. Schwarz, Schiffer, Wolf) genutzt werden [BKPZ06]. Zudem wird der Gebärdennamen zusätzlich durch das Mundbild des amtlichen Vor- oder Nachnamens ergänzt [Uhl12].

Bei der Analyse sprachlicher Äußerungen ist demnach zu beachten, dass sich die Namengebärde auf ein Individuum bezieht. In dem vorliegenden Beispiel steht die verwendete Glosse PETER also für den Gebärdennamen der anwesenden Person und nicht allgemein für den Vornamen *Peter* oder für das ursprüngliche Lexem aus dem diese Namengebärde gebildet wurde. Die Glossenbezeichnung „PETER“ wurde hier so gewählt, dass ein Bezug zu dem lautsprachlichen Vornamen der gemeinten Person hergestellt werden kann. Für die Interpretation erfolgt hier vereinfacht die Zuordnung zu dem männlichen Vornamen *Peter*. Auf diese Weise ist eine Vergleichbarkeit der Interpretationsergebnisse von Laut- und Gebärdensprache gewährleistet.

5.2 Sprachmaterial zur semantischen Modellierung

Eine *Elementargrammatik* zu der im Abschnitt 5.1 beschriebenen Bedeutung M stellt die Grundlage für die weiteren Modellierungsschritte im gewählten Ansatz dar. Voraussetzung für die Erstellung dieser Grammatik ist eine geeignete Sprachmaterialsammlung. Die gesprochene Sprache ermöglicht eine Vielzahl unterschiedlicher Realisierungen, um eine konkrete Bedeutung auszudrücken. Idealerweise wird die Menge möglicher Sprachäußerungen $\mathcal{U}(M)$ zu festgelegten *semantischer Ankern* empirisch ermittelt. Allgemein kann jedoch nicht ausgeschlossen werden, dass eine Teilmenge von $\mathcal{U}(M)$ unbeobachtet bleibt. Mit Hilfe dieser Sammlung sprachlicher Äußerungen ist auch die Identifikation von sogenannten *synonymischen Syntagmen* (bedeutungsgleiche Folgen von sprachlichen Zeichen) möglich. Für Neuentwicklungen ist die Durchführung von *Wizard-of-Oz Experimenten* zweckmäßig. Zudem ist auch eine konkrete Befragung von Probanden üblich, welche z. B. mögliche Variationen sprachlicher Realisierungen zum Erreichen eines Kommunikationsziels benennen oder aufschreiben sollen. Zusätzlich können aber auch Daten aus real-existierenden Anwendungen, wie zum Beispiel Sprachdialogsystemen oder zwischenmenschlich geführte Telefongespräche verwendet werden.

Wizard-of-Oz Experimente zur Sammlung von Sprachdaten

Der Begriff *Wizard-of-Oz* basiert auf den bekannten Kinderbuchtitel des amerikanischen Autors LYMAN FRANK BAUM „*The Wonderful Wizard of Oz*“. Durch die Veröffentlichungen in den 1980er Jahren von JOHN F. KELLEY (u. a. [Kel83, Kel84]) hat sich der Terminus *Wizard-of-Oz Experiment* in dem Bereich der Mensch-Maschine-Interaktion etabliert. Er wird heute für Verfahren verwendet, bei dem die Interaktion zwischen einem Benutzer und dem technischen System simuliert wird. Hierbei übernehmen vom Testteilnehmer separierte Personen die Funktionen der Maschine. Die Probanden hingegen werden in dem Glauben gelassen, dass sie mit einem funktionierenden System interagieren. Diese Methode wird sowohl für die Gewinnung natürlichsprachlicher Informationen [FG91] als auch für multimodale Anwendungen [RAL⁺06] genutzt. Ein bekanntes Beispiel stellt der ATIS-Korpus dar. Dazu wurde ein automatisiertes Sprachdialogsystem simuliert. Neben konkreten Datenbankabfragen zu Fluginformationen waren auch Erkundigungen über Airlines, Tarife und Flughäfen möglich [Pri90]. Auch für französischsprachige Datensammlung des MEDIA-Projekts [BMB⁺06, BMQD09] wurde ein Touristeninformationssystem vorgetäuscht. Hier konnte der Proband lautsprachlich Hotelreservierungen vornehmen.

Solche Simulationen sind oft kostengünstiger und schneller zu realisieren als die Implementierung kompletter Prototypen. Trotzdem sind Entwicklung und Durchführung von *Wizard-of-Oz Experimenten* aufwändig und zeitintensiv.

Für eine Bestimmung von möglichen sprachlichen Verhalten zu der Bedeutung M

muss der Proband in die intendierte Interaktionssituation (festgelegt durch *semantische Anker*) „hineinversetzt“ werden. Eine wichtige Rolle spielt dabei die Gestaltung der Aufgabenstellung in *Wizard-of-Oz Experimenten*. Für natürlichsprachliche Dialogsysteme ist es beispielsweise sinnvoll, dass Aufgabenstellungen bevorzugt Grafiken und Symbole beinhalten. Bei sprachlichen Anweisungen besteht die Gefahr, dass diese Einfluss auf die Wortwahl der Probanden bei der Artikulation haben. Zudem hat sich in [Stö14] gezeigt, dass die syntagmatischen Relationen der auf den Aufgabenbeschreibungen angeordneten Grafiken auch die syntaktische Struktur der gewählten Sprachäußerung beeinflusst. Ein vielversprechender Ansatz scheint die Gestaltung nach dem ABC-Schema (vgl. Kapitel 4.2) zu sein. Dabei ist der aktuelle Zustand (*antecedents*) und der gewünschte Zielzustand (*consequences*) dargestellt. Der Proband muss durch geeignetes sprachliches Verhalten das technische System dazu bringen, dass dieses die visualisierte „Soll-Situation“ einstellt. Mit Hilfe der protokollierten Daten aus der simulierten Interaktion ist es bereits in der Entwicklungsphase möglich, systematisch das Systemdesign, die Bedienbarkeit und auch den Sprachdialogablauf zu evaluieren und iterativ zu optimieren. Entsprechende Erläuterungen dazu findet man u. a. in [GWH85, DJA93, BDD98, Mö10]. Für das Beispiel einer Home-Entertainment-Anwendung wurde in [KRRL12, Kar12] das System statisch und dynamisch modelliert. Dazu sind entsprechende Systemstrukturen und -funktionen definiert worden. Anhand der dafür gesammelten Sprachdaten wurde bereits in [RW14, Rit13, HKLW13] die Konstruktion von Transduktoren zur semantischen Interpretation untersucht.

Generierung von Sprachäußerungen zur Elementargrammatik der Beispielanwendung

Für die praktischen Untersuchungen wurde die Generierung von Sprachdaten mit Hilfe der *Wizard-of-Oz Methode* nicht genutzt. Ein wesentlicher Faktor stellt das ungünstige Verhältnis von Aufwand und Nutzen dar. Problematisch in diesem Zusammenhang ist neben dem relativ kleinen Umfang (Beschränkung auf nur ein *semantisches Schema*) vor allem die Gewinnung von gehörlosen Probanden. Hierfür ist es sinnvoll, dass ein Mitglied des Forschungs- oder Entwicklungsteams die Deutsche Gebärdensprache beherrscht. Die gewünschten Systemreaktionen können auf diese Weise sicher ausgewählt werden. Ein weiterer Vorteil ist die Möglichkeit einer schnellen und unkomplizierten Beantwortung von eventuell auftretenden Rückfragen der Versuchspersonen vor Ort. Die genutzten Sprachäußerungen sind für diese Arbeit lediglich dadurch entstanden, dass möglichst viele denkbare Sprachzeichenfolgen zu der im Abschnitt 5.1 beschriebenen Bedeutung selbst überlegt und notiert wurden. Solch ein Vorgang sollte vorzugsweise von Muttersprachlern der verwendeten Sprache oder Personen mit ausreichend Sprachkompetenz durchgeführt werden. Aus diesem Grund wurden die Gebärdensprachäußerungen von der *Fachgruppe Gebärdensprachdolmetschen* an der *Westfälischen Hochschule Zwickau* artikuliert und transkribiert. Die erstellten Transkriptionen bilden die Grundlage für

die Glossennotationen, welche zur Modellierung der Grammatiken verwendet werden.

Das verwendete Sprachmaterial

Benutzeräußerungen der deutschen Lautsprache

Für das in Kapitel 5.1 beschriebene Theoriebeispiel „*Dort drüben steht Peter und wartet.*“ sind 96 lautsprachliche gleichbedeutende Benutzeräußerungen erstellt worden. Grundlage dafür sind 24 verschiedene Formulierungen, bei denen die relative Ortsangabe nicht lexikalisiert wurde. Für jede dieser 24 Sprachäußerungen werden jeweils 4 sprachliche Varianten für das *semantische Konzept* eingesetzt. Dies sind die in diesem Kontext möglichen Synonyme: *da*, *dort*, *da drüben* und *dort drüben*. Berücksichtigt wurde auch die Möglichkeit der Verwendung von Partizipien in der deutschen Sprache (hier: *stehend*, *wartend*) für den Ausdruck von Aktionen. Teilweise ist auch die Verwendung des Artikels „*der*“ möglich, ohne dass sich die intendierte Absicht verändert. Somit sind z. B. auch die Formulierungen „*Der wartende Peter . . .*“ oder „*Dort drüben steht der Peter und wartet*“ möglich. Ein vollständiger Überblick zu den verwendeten *synonymische Syntagmen* der lautsprachlichen Äußerungen findet man im Anhang A.1.

Zusätzlich sind hier weitere bedeutungsähnliche sprachliche Ausdrücke denkbar. Eine Benutzeräußerung kann beispielsweise auch aus mehreren Sätzen bestehen. Im Anhang A.2 sind deshalb Aussagen mit 2 Sätzen aufgeführt. Eine mögliche sprachliche Realisierung könnte „*Peter steht da drüben. Peter wartet.*“ sein. Dies führt zu auftretenden Wiederholungen von einzelnen Wörtern. Die Sätze einer Äußerung können wiederum durch das Wort „*und*“ verknüpft werden.

Eine zusätzliche Erweiterungsmöglichkeit ist hier auch die Einführung des Personalpronomens „*er*“. Diese anaphorische Verbindung verweist auf die bereits zuvor genannte Person „*Peter*“. Somit könnte dann die Äußerung „*Peter steht da drüben. Er wartet.*“ auch einen vergleichbaren semantischen Gehalt besitzen. Die ergänzend aufgeführten Beispieläußerungen (vgl. Anhang A.2) wurden aufgrund der zusätzlichen Bedeutungseinheit bzw. durch die Wiederholungen von sprachlichen Zeichen nicht für die *Elementargrammatik* berücksichtigt. Diese Entscheidung darf nicht als allgemeingültig betrachtet werden. In einem kognitiven System gehören auch diese zusätzlichen Varianten sprachlichen Verhaltens möglicherweise zu derselben kommunikativen Absicht *M*. Die zusätzlich erstellten sprachlichen Realisierungsvarianten sind jedoch Bestandteil der Trainingsdaten für das verwendete stochastische Sprachmodell bei der Artikulation. Weiterhin sind diese auch für die Erstellung des stochastischen *konzeptuellen Sprachmodells* zur semantischen Interpretation (siehe Kapitel 6.1.2) verwendet worden.

Sprachäußerungen der Deutschen Gebärdensprache

Grundlage für die Analyse der Gebärdensprachäußerungen sind die Transkriptionen der an der *Westfälischen Hochschule Zwickau* erstellten Gebärdensprachaufzeich-

nungen. Insgesamt wurden 21 Sprachäußerungen verschriftlicht (vgl. Anhang A.4). Die Gebärdensprachtranskriptionen umfassen dabei neben den Glossen zur manuellen Gebärdendarstellung auch die beiden nichtmanuellen Parameter *Mundbild* / *Mundgestik* und *Mimik* des Sprechers (vgl. Kapitel 2.2).

Exemplarisch ist hier die Notation des Sprachbeispiels 8 durch Abb. 5.2 gegeben. Dabei markiert das Symbol „Ao“ die gehobene Augenbrauen und die Markierung „[Zz]“ die gezeigten Zähne des Sprechers während der manuellen Artikulation der Gebärde „DORT“. Zudem steht hier das Notationszeichen „...“ für eine Sprechpause. Alle

Mimik:	neutral	Ao	neutral
Hände:	PETER DORT STEHEN ... WARTEN		
Mundbild:	peter	[Zz]	warten

Abb. 5.2: Beispiel einer Gebärdensprachtranskription

Gebärden wurden im bedeutungsneutralen Raum vor dem Oberkörper gebärdet. In den Gebärdensprachäußerungen wurde die spezielle Namengebärde für Peter (PETER) durch die allgemein verständliche Gebärde MÄDCHEN ersetzt. Für die Erstellung der Grammatiken kann die Glosse MÄDCHEN allerdings wieder problemlos durch die Glosse PETER substituiert werden.

Die Auswertung der Gebärdensprachtranskriptionen ergab, dass nur etwa die Hälfte der Gebärdensprachäußerungen für die Erstellung der *Elementargrammatik* genutzt werden können. In fünf Transkriptionen fehlte der explizite Ausdruck der relativen Ortsangabe. Ein Grund hierfür könnte sein, dass sich dieser Aspekt bereits durch die Gesprächssituation ergibt. Möglicherweise wurde allerdings diese Information vom Sprecher auch als zweitrangig eingestuft und somit auch nicht artikuliert. Weitere fünf Gebärdensprachbeispiele wurden nicht genutzt, da diese einem vergleichbaren Spracherkenntnis zuzuordnen sind.

Die in den Transkriptionen enthaltenen Unterschiede in der Beschreibung von Mimik, Mundbild und Pausengestaltung haben meist keinen Einfluss auf das Interpretationsergebnis. So unterscheiden sich die Beispielsätze 3, 4, 9 und 20 des Anhangs A.4 lediglich durch Mundbild und Pausengestaltung. In der DGS werden Pausen bewusst zur syntaktischen Gliederung und für die Markierung von Satzverbindungen eingesetzt [PMMH08]. Dies beeinflusst natürlich auch den Rhythmus der Gebärdensprache. Prosodische Informationen und damit auch mögliche Satzzeichen werden in dem betrachteten technischen System auch für die Lautsprache nicht berücksichtigt. Mundbilder und Mundgestiken werden in dieser Arbeit als Bestandteil des Gebärdensprachzeichens betrachtet. Die im Wörterbuch vorhandenen Lexeme beinhalten also bereits das Mundbild. Dementsprechend würde dieselbe manuelle Gebärde bei Verwendung verschiedener bedeutungsunterscheidener Mundbilder jeweils auch zu unterschiedlichen Glossen führen. Dasselbe gilt für die verwendete sprachliche Mimik, welche oft die Bedeutung einer Gebärde unterstützt und zudem auch eine prosodische Funktion besitzt. Damit können

z. B. auch besonders wichtige Aspekte eine Sprachäußerung betont werden. Jedoch kann der Gesichtsausdruck auch zusätzliche Informationen liefern, welche ggf. von einem sprachverarbeitenden System erfasst werden müssen. So besteht die Möglichkeit, einer manuellen Gebärde durch Mimik eine adjektivische Bedeutung hinzuzufügen. Zudem spiegelt nichtsprachliche Mimik auch die Emotionen des Sprechers wider. Eine Beson-

Mimik:	neutral	Ao	neutral
Hände:	PETER STEHEN ... DORT ... STEHEN		
Mundbild:	peter	[Zz]	

Abb. 5.3: Gebärdensprachtranskription der Beispielaussage 6

derheit stellt die Transkription der Gebärdensprachäußerung 6 dar, welche in Abb. 5.3 dargestellt ist. In dieser Sequenz kommt wiederholt die Gebärde „STEHEN“ vor, jedoch kein Ausdruck des Wartens. PROF. ROSENSTOCK vermutet darin die Hervorhebung des Stehens in der Äußerung. Gleichzeitig merkt sie an, dass solche in der Gebärdensprache auftretenden Reduplikationen bisher unzureichend erforscht sind [Ros15]. Möglicherweise könnte auch die wiederholte Artikulation der Aktion „stehen“ eine längere Dauer des Stehens ausdrücken, was u. U. auch als „warten“ interpretiert werden könnte. Aus diesem Grund findet die Beispielaussage 6 auch in dem Transduktor zur *Elementargrammatik* Berücksichtigung. Da dieser Prozess möglicherweise nicht auf alle weiteren Aktionen anzuwenden ist, wird dieses Sprachbeispiel für die *mikrolokale Grammatik* nicht verwendet. Somit werden letztendlich 11 Gebärdensprachäußerungen für die Erstellung der *Elementargrammatik* genutzt. Davon werden 10 Glossenfolgen zur Modellierung des *Äußerungs-Bedeutungs-Transduktors* zur *mikrolokalen Grammatik* weiterverwendet. Eine vollständige Auflistung der verwendeten Glossennotation ist im Anhang A.3 zu finden.

Anmerkungen zu Unterschieden im gesammelten Sprachmaterial von Laut- und Gebärdensprache

Bei einer vergleichenden Untersuchung der Sprachäußerungen von Laut- und Gebärdensprache sind neben den im Kapitel 2.2 beschriebenen prinzipiellen Unterschieden weitere Abweichungen zwischen den Sprachzeichenfolgen festzustellen. Hier sind vor allem die in den Gebärdensprachtranskriptionen auftretenden zusätzlichen Glossen zu nennen. Diese können u. U. sogar zusätzliche semantische Einheiten darstellen. Die Gebärden WO und WER in den Beispielen 7 und 10 des Anhangs A.3 beziehen sich hier direkt auf die Orts- bzw. Personenangaben. Dabei sind sie den semantischen Einheiten DORT und PETER direkt vorangestellt. Sie stellen somit Indikatoren für das jeweilige semantische *Konzept* dar und werden diesem auch zugeordnet.

Differenzierter muss die Verwendung der Glosse SELBST betrachtet werden. In den Sprachäußerungen 8 und 9 wird diese Gebärde direkt vor der Namensgebärde *PETER*

artikuliert. Damit wird direkt auf den Akteur der vorab beschriebenen Handlungen *stehen* und *warten* hingewiesen. Die Ausführung der Gebärde SELBST kann im Allgemeinen variieren und somit sowohl auf die eigene Person (*ich*), als auch auf andere Personen (*du, er, sie*) bezogen sein. In dem Beispielsatz 6: „DORT PETER STEHEN SELBST WARTEN“ könnte allerdings die Glosse SELBST als Relativpronomen aufgefasst werden. Solche Sprachbeispiele wurden bei den lautsprachlichen Äußerungen zur *Elementargrammatik* allerdings nicht verwendet. Ähnlich zum DGS-Beispielsatz 6 wäre z. B. das Sprachbeispiel „Dort drüben steht Peter. Dieser/Der/Er wartet (dort).“ anzusehen.

Durch das Anhängen eines „d“ an die Grundform des Verbes (hier: *stehend, wartend*) in der deutschen Lautsprache erhält die Verbform die Bedeutung eines Adjektivs. In der Deutschen Gebärdensprache gibt es solche morphologischen Prozesse hingegen nicht [Ebb15]. An dieser Stelle ist auch anzumerken, dass es zu den Differenzierungen innerhalb der aufgeführten lautsprachlichen Äußerungen „meist keine unmittelbaren syntaktischen Entsprechungen in der DGS“ [Ebb15] gibt.

5.3 Modellierung von Äußerungs-Bedeutungs-Transduktoren

Dieses Kapitel beschreibt einen möglichen Ablauf zur Erzeugung von *Äußerungs-Bedeutungs-Transduktoren* zu *semantischen Schemata*.

5.3.1 Erstellung eines Äußerungs-Bedeutungs-Transduktors zur Elementargrammatik

Auf Basis der gesammelten Sprachäußerungen zu der *Elementargrammatik* (vgl. Abschnitt 5.2) ist ein *Äußerungs-Bedeutungs-Transduktor* zu erstellen. Dieser erzeugt zu der Menge akzeptierter Sprachäußerungen $\mathcal{U}(M)$ eine *MWR-Zeichenkette*. Für die



Abb. 5.4: Graph des Transduktors zur DGS-Beispieläußerung 5

Verarbeitung von erkannten Gebärdensprachäußerungen durch endliche Automaten ist eine geeignete Glossennotation notwendig (vgl. Kapitel 2.2.3). Diese muss die simultan auftretenden Gebärdensprachbestandteile auf eine beschriftete Totalordnung abbilden.

Im ersten Schritt erfolgt eine Formatierung der Menge von gesammelten Sprachzeichenfolgen. Hierfür werden Umlaute sowie das Schriftzeichen „ß“ ersetzt. Zusätzlich sind eventuell vorhandene Satzzeichen zu entfernen. Bei den Gebärdensprachäußerungen ist darauf zu achten, dass keine Interpunktionszeichen gelöscht werden, die Bestandteil der Glossennotation sind. Die Erstellung des Transduktors zu der *Elementargrammatik* ist durch folgende Schritte realisiert worden:

1. Erzeuge zu jeder Sprachäußerung einen endlichen Automaten T_{Si} , welcher an einem eingelesenen bedeutsamen Wort die dazugehörige semantische Einheit (Attribut-Wert-Paar) zuordnet (vgl. Abb. 5.4). Für die zu erzeugenden *Konzepte* ist die Syntax der *MWR-Zeichenketten* zu berücksichtigen.
2. Erzeuge iterativ einen Gesamtautomaten T_{EG} durch die Summe (Union) zuvor erstellter Transduktoren T_{Si} : $T_{EG} = \bigoplus_{i=1}^N T_{Si}$.
3. Optional kann ein *äquivalenter*, jedoch topologisch optimierter Transduktor generiert werden. Der in der *OpenFst*-Bibliothek verwendete Minimierungsalgorithmus setzt einen deterministischen Transduktor voraus. Aus diesem Grund hat sich die Reihenfolge I. *Determinierung* II. *Minimierung* und III. *Entfernung von Epsilon-Transitionen* bewährt. Der optimierte Transduktor ergibt sich dementsprechend durch $T_{EGopt} = \xi(\min(\det T_{EG}))$.
4. Einsetzen des (optimierten) Gesamttransduktors in einen „Root-FST“ zum übergeordneten semantische Schema. Dieser dient zur Generierung einer syntaktisch korrekten *MWR-Zeichenkette*.



Abb. 5.5: Graph des „Root-FST“ zur Erstellung eines endlichen Äußerungs-Bedeutungs-Transduktors zum semantischen Schema „PERS-ACT-LOC“

Der „Root-FST“ generiert in diesem Fall die initiale *MWR-Zeichenkette* „FVR[PERS-ACT-LOC<mwr>]“. Durch den letzten Schritt wird in diesem Automaten die Kante mit dem Symbol <mwr> durch den Gesamttransduktor zur *Elementargrammatik* ersetzt. Dazu steht in *OpenFst* die Automatenoperation *Replace* (*Ersetzung*) zur Verfügung. Der Graph des für die *Ersetzung* verwendeten initialen Transduktors ist in Abb. 5.5 dargestellt.

Als Ergebnis dieser Substitution liegt für die deutsche Lautsprache und für die Deutsche Gebärdensprache jeweils ein *Äußerungs-Bedeutungs-Transduktor* vor. Abb. 5.6 zeigt den Graphen eines solchen Automaten. Das Eingabealphabet besteht aus den Wörtern bzw. Glossen der jeweiligen Quellsprache und möglichen Sondersymbolen, wie beispielsweise dem leeren Eingabesymbol <eps>. Ausgabealphabet für Laut- und

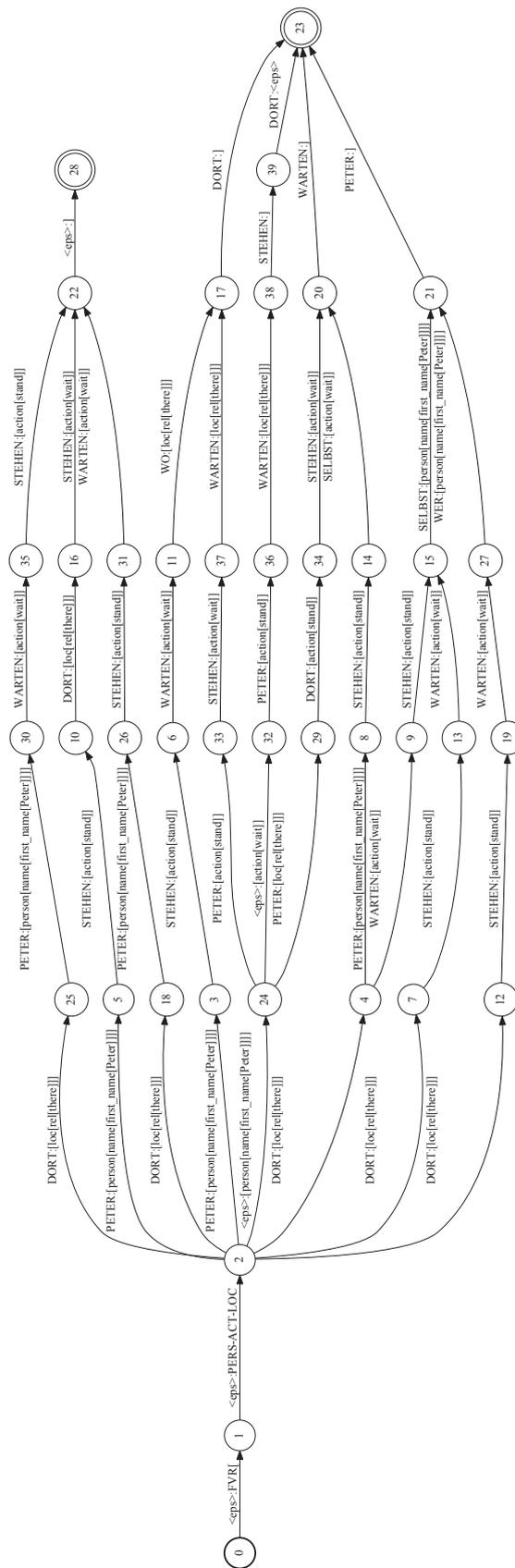


Abb. 5.6: Automatengraph des endlichen Äußerungs-Bedeutungs-Transduktors zur gepärdensprachlichen Elementargrammatik

Gebärdensprache sind hierbei identisch. Hier wurden die Lexikoneinträge so gewählt, dass diese *semantische Anker* in der Klammerstruktur einer *MWR-Zeichensequenz* beschreiben. Damit erzeugt der erstellte endliche Automat letztendlich eine syntaktisch korrekte *MWR-Zeichenkette* zu einem durchgehenden Pfad π .

5.3.2 Platzhalter in mikrolokalen Grammatiken

Die bisher erstellte semantische Grammatik ordnet einer akzeptierten Sprachäußerung lediglich zwei konkrete Bedeutungsstrukturen zu. Im Kapitel 5.1 wurde festgelegt, dass diese *Merkmal-Werte-Relationen* zu einer Bedeutung M gehören. In der *Elementargrammatik* sind bereits Synonyme und verschiedene Wortstellungen berücksichtigt worden. Wichtig dabei ist, dass alle Variationen der akzeptierten sprachlichen Realisierungen zu derselben Bedeutung M gehören. Hierbei sind neben den aufgeführten Attributen auch die Datenwerte fest vorgegeben. Wir erhalten somit für alle Sprachäußerungen, welche zu dieser *Elementargrammatik* passen, dieselbe *Merkmal-Werte-Relation*. Für den praktischen Einsatz sind *Äußerungs-Bedeutungs-Transduktoren* zu einer *Elementargrammatik* ungeeignet. Aus diesem Grund ist im nächsten Schritt ein Transduktor zu einer *mikrolokalen Grammatik* zu erstellen. Wie im Kapitel 4.2.2 beschrieben, gehört diese zu einem *semantischen Schema* S . Der Grad der Abstraktion ist dabei nicht spezifiziert und kann je nach Modellierung unterschiedlich ausgeprägt sein. So können selbstverständlich auch *semantische Schemata* Bestandteil eines übergeordneten *semantischen Schemas* sein. Je nach Art der Anwendung sind bei der Modellbildung *semantische Anker* variabel zu gestalten. Die ausgewählten Einheiten werden im *semantischen Schema* durch Platzhalter ersetzt. Diese freien Parameter können

- variable Datenwerte bzw. Ausprägungen eines konkreten Attributes,
- verschiedene Ausprägungen von Merkmalen mit dazugehörigen Werten oder
- untergeordnete semantische Schemata

sein. Für die Platzhalter existieren also unterschiedliche Modellierungstiefen. Die in einigen aktuellen stochastischen Modellen verwendeten Wortklassen können als Platzhalter für Datenwerte betrachtet werden. Für die Modellierung von Platzhaltern ist eine direkte Zuordnung von einzelnen Zeichen einer Sprachäußerung zu *semantischen Ankern* nötig. Als Beispiel für einen Platzhalter dient das Objekt *person*, welches mehrere Attribute besitzt. In diesem Fall führt ein Platzhalter für verschiedene Vornamen (= Datenwerte) des Attributs *first_name* zu dem variablen *semantischen Anker* `[person[name[first_name[v]]]]`.

5.3.3 Modellierung eines Äußerungs-Bedeutungs-Transduktors zur mikrolokalen Grammatik

Für die Realisierung des *Äußerungs-Bedeutungs-Transduktors* zum *semantischen Schema* werden sowohl für die deutsche Lautsprache als auch für die Deutsche Gebärdensprache drei Generalisierungen verwendet. Dies betrifft konkret die Platzhalter für auszufüh-

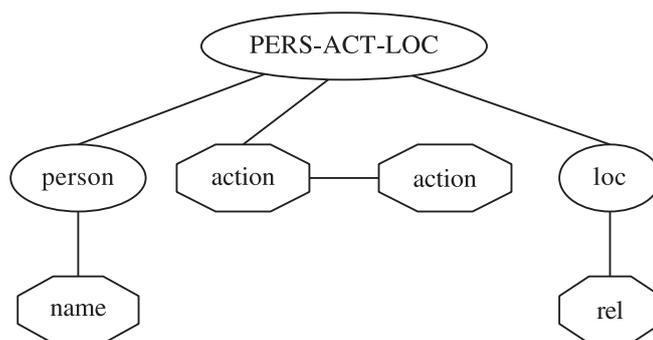


Abb. 5.7: Verbindliche Merkmale des modellierten semantischen Schemas

rende Aktionen, relative Ortsangaben und verschiedene Namensbezeichnungen von Personen. Abb. 5.7 zeigt die festgelegten semantischen Kategorien zu der *Merkmals-Werte-Relation* des *semantischen Schemas* „PERS-ACT-LOC“. Ein Platzhalter ist in dieser unspezifizierte Referenzstruktur als Oktagon dargestellt. Bei der Interpretation einer Sprachäußerung werden diese Platzhalter spezifiziert. In Abb. 5.7 ist ersichtlich, dass zwei Aktionen zu dem *semantischen Schema* gehören. In diesem Fall soll auch die Reihenfolgeinformation der Aktionen in der *Merkmals-Werte-Relation* Berücksichtigung finden. Alternativ wären auch zwei unterschiedliche Aktionsbezeichnungen möglich. Zu diesen könnten dann jeweils verschiedene Wertemengen gehören. Neben der Berücksichtigung verschiedener Werte in den Eingabesymbolen, ist hier auch die zu erzeugende Klammerstruktur der *MWR-Zeichenkette* in die Überlegungen zur Gestaltung der endlichen Transduktoren einzubeziehen.

Prinzipiell kann ein *Äußerungs-Bedeutungs-Transduktor* durch folgende Arbeitsschritte realisiert werden:

1. Erstellung eines initialen *Äußerungs-Bedeutungs-Transduktors* zu einer *Elementargrammatik* (vgl. Kapitel 5.3.1).
2. Erzeugung von endlichen Transduktoren für die modellierten Platzhalter. Diese müssen alle möglichen Werte beinhalten und alternative sprachliche Realisierungen berücksichtigen.
3. Die Ausgabesymbole (semantische Einheiten) im initialen Transduktor werden durch das dazugehörige Nichtterminalsymbol des assoziierten Platzhalters ersetzt. Die dazugehörigen Sprachzeichen der Eingabephase sind durch das leere Symbol zu substituieren.

4. Ersetzung (*OpenFst: Replace*) der Platzhaltersymbole an den Kanten durch die dazugehörigen Transduktoren.
5. Optimierung des erzeugten *Äußerungs-Bedeutungs-Transduktors* T_{UM} .

Zu beachten ist hierbei, dass sich der Wert eines Attributes teilweise aus einer Folge von Wörtern bzw. Glossen ergibt. Demnach ist auch eine Betrachtung der lexikalischen Bedeutung der Sprachzeichen bei der Modellierung von Grammatiken nötig. Platzhalter können auch mehrere Attribute und sogar ganze semantische Schemata umfassen. Die Automaten dieser Platzhalter sind teilweise sehr komplex. Aus diesem Grund ist die Zerlegung des Problems in kleinere Teilprobleme zu empfehlen. So bleibt die Modellbildung besser beherrschbar.

Der Detaillierungsgrad der erstellten Platzhalterautomaten für Laut- und Gebärdensprache ist teilweise unterschiedlich. Bei der Modellierung der Platzhalter ist zu beachten, dass jeder Platzhaltertransduktor durch die erzeugte Klammerstruktur einen *semantischen Anker* beschreibt. In dieser Anwendung definiert der Automat zu einem Platzhaltersymbol alle möglichen Bedeutungsstrukturen zu einem Kindknoten der Wurzel des unkaskadierten Baumes eines *semantischen Schemas*. Dies hat zur Folge, dass hier beispielsweise alle vom Platzhalter *name* generierten *MWR-Zeichenketten* auch den übergeordneten Entitätstyp *person* beinhalten.

Platzhalter *loc_rel*

Im Platzhalter für die relative Ortangabe sind zwei verschiedene Datenwerte berücksichtigt worden. Zusätzlich zu $\mathbf{v} = there$ kann dem Merkmal *loc_rel* auch der Wert $\mathbf{v} = here$ zugeordnet werden. In dem fest vorgegebenen Kontext des semantischen Schemas besteht in diesem Fall also eine *paradigmatische Relation* zwischen den beiden *Konzepten*. Das *Konzept* $[loc[rel[there]]]$ ist dem Eingabewort *hier* bzw. den Glossen *HIER* und

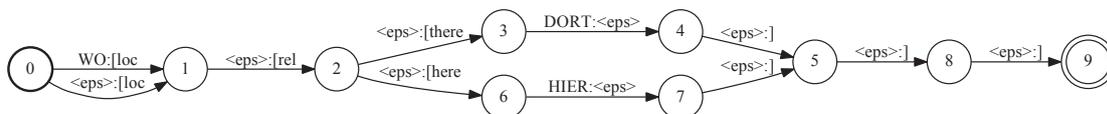


Abb. 5.8: FST für Platzhalter „loc_rel“ (DGS)

WO HIER zugeordnet. Eine Besonderheit stellt das in den Gebärdensprachebeispielen verwendete Fragedeterminativ *WO* dar. Die Glosse *WO* stellt hier einen Ortsbezug her und kann deshalb der übergeordneten semantischen Kategorie *loc* zugeordnet werden. Abb. 5.8 zeigt exemplarisch für die Gebärdensprache den Automatengraphen des Transduktors zum Platzhalten *loc_rel*. Zum Vergleich ist der Graph für den lautsprachlichen Platzhalterautomaten im Anhang B dargestellt.

Platzhalter *action*

Die Handlungen „stehen“ ($\mathbf{v} = stand$) und „warten“ ($\mathbf{v} = wait$) sind in den *Äußerungs-*

Bedeutungs-Transduktoren für Laut- und Gebärdensprache dem Attribut *action* zugeordnet. Der Transduktor zur *mikrolokalen Grammatik* soll auch weitere Tätigkeiten akzeptieren können. Aus diesem Grund wurden exemplarisch „rauchen“ ($\mathbf{v} = \text{smoke}$), „liegen“ ($\mathbf{v} = \text{lie}$) und „sitzen“ ($\mathbf{v} = \text{sit}$) als zusätzlich mögliche Aktionen aufgenommen. Für die Lautsprache werden ein großer Teil möglicher Verbflexionen berücksichtigt und in dem Modell zusammengefasst. Die Konjugationen durch Numerus, Tempus und Partizip führen hier zu keiner Veränderung der Bedeutung des sprachlichen Verhaltens. Sie gehören jeweils zu demselben Lexem. Zum Beispiel enthält die Wortklasse „STEHEN“, welche zu dem Datenwert $\mathbf{v} = \text{stand}$ zugeordnet wird, folgende Elemente:

Präsens - *stehe, stehst, steht, stehen*

Imperativ - *steh*

Präteritum - *stand, standest, standen, standet*

Partizip - *stehend, stehende, stehenden*

Nicht berücksichtigt werden hier allerdings die zusammengesetzten Zeitformen Futur, Perfekt sowie Plusquamperfekt.

Auf diese Weise können Automaten für entsprechende Handlungen erzeugt werden, welche unverändert auch für andere semantische Schemata nutzbar sind. Die Abb. B.3 im Anhang B zeigt den Automatengraph des Platzhalters *action* für die deutsche Lautsprache. Einfache, körpergebundene Verben (z. B. RAUCHEN, WARTEN) werden in der Deutschen Gebärdensprache (DGS) nicht flektiert. Eine Flexion von Verben innerhalb der DGS kann bei orts- oder personenkongruenten Verben auftreten. Hierzu zählen auch die verwendeten *Raumverben* STEHEN, SITZEN und LIEGEN. Die Ausführungsstelle der Gebärde kann hier Informationen zu der Position eines Lebewesens oder Objekts liefern, auf die sich die Handlung bezieht [EHH12].

Insofern können hier zu einer Glosse AAA ergänzende Angaben über die Position im Gebärdenraum hinzugefügt worden sein. Wenn die Gebärdenspracherkennung die Verortung bei der Klassifikation unterscheidet, so wären auch Notationen der Form AAA-(*loc*), wie z. B. STEHEN-*li* oder STEHEN-*re*, als Klassifikationsergebnis eines Gebärdenspracherkenners möglich. Anders als bei der Lautsprache wurde für den DGS-

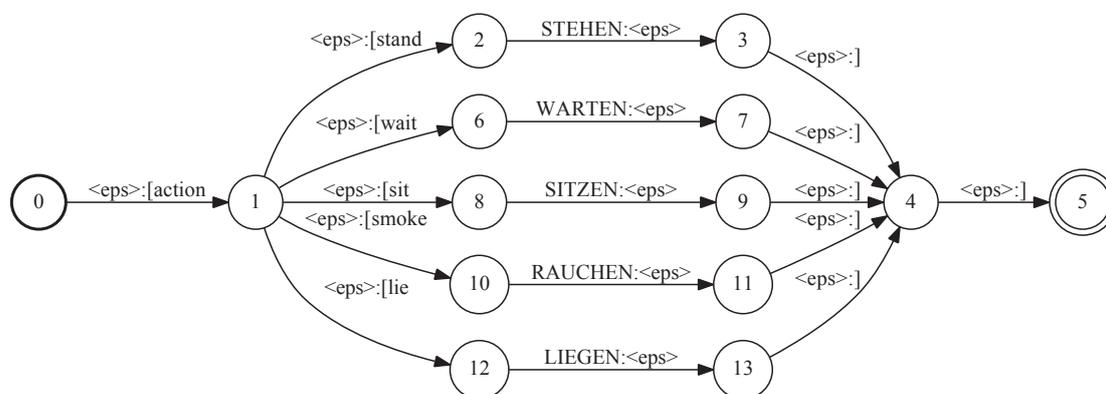


Abb. 5.9: FST für Platzhalter „action“ (DGS)

Platzhalter „action“ keine Verbflexionen berücksichtigt. Somit enthält der in Abb. 5.9 gezeigte Automatengraph jeweils nur eine Glosse.

Platzhalter *name*

Für die eigenen Untersuchungen wurde der Platzhalter auf den Namen einer Person eingeschränkt. Dieser kann aus Vornamen und Nachnamen zusammengesetzt sein. Bei der Modellierung des Platzhalters sind jeweils auch mögliche Doppelnamen berücksichtigt worden. Für diese Generalisierung ist natürlich eine Vielzahl von Werten möglich. Hier wurden je drei Vor- und Nachnamen verwendet. Dabei sind nur die Sprachzeichen *Sabine* und *Peters* einem Attribut eindeutig zugeordnet. Die Bezeichnungen *Steffen* und *Peter* können hingegen sowohl Vor- als auch Familienname sein. Bei *Peter* wurde der Familienname beispielsweise aus dem Vornamen des Vaters abgeleitet. Neben *Peters*, *Petersen* und *Peterson* gehört nämlich auch der Nachname *Peter* zu den patronymischen Formen des Vornamens *Peter*. Mögliche Namenszusätze, wie z. B. akademische Grade und Amtstitel, wurden bei der Modellierung der Grammatik dieser Beispielanwendung nicht berücksichtigt. Sie sind aber lexikalische Indikatoren für den Entitätstyp *person*. Der Transduktor für die Generalisierung *name* verwendet für Laut- und Gebärdensprache die Automaten für die Platzhaltersymbole `<first_name>` und `<last_name>`. In dem zu erstellenden *Äußerungs-Bedeutungs-Transduktor* werden die Platzhaltersymbole dann mit dem jeweiligen FST (siehe Abb. 5.10 und 5.11) ersetzt. Exemplarisch wird für die Lautsprache der übergeordnete initiale Transduktor für Platzhalter *name* in Abb. 5.12 verwendet. Zusätzlich zu dem eigentlichen Namen können Wörter vorangestellt sein, welche auf das Merkmal hinweisen. Hier wurden beispielhaft die Anredebezeichnungen *Herr* und *Frau* sowie bestimmte Artikel in den Platzhalter aufgenommen. Diesen sprachlichen Zeichen ist in dem verwendeten Modell kein eigenständiges *Konzept* zugeordnet worden. Durch die Berücksichtigung der Artikel *der* und *die* beim Merkmal *person* sind die für die *Elementargrammatik* verwendeten lautsprachlichen Beispieläußerungen

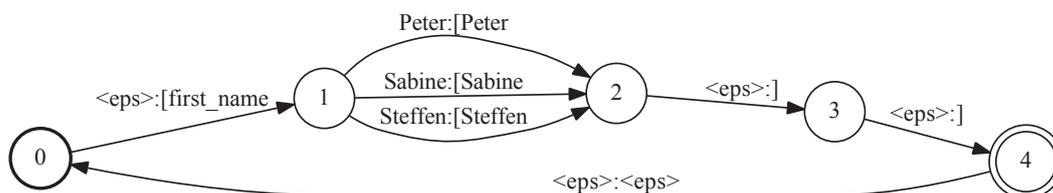


Abb. 5.10: FST für Vornamen (Lautsprache)

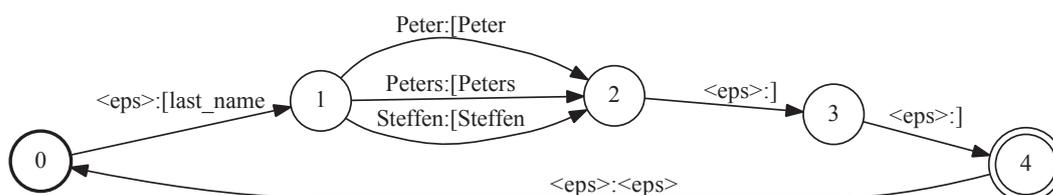


Abb. 5.11: FST für Familiennamen (Lautsprache)

15 bis 24 im Anhang A.1 redundant. Der Artikel *der* tritt in den lautsprachlichen Äußerungen auch alleinstehend vor den Partizipien auf. In diesem Fall wurde der Artikel in dem Transduktor zur *Elementargrammatik* durch den separaten Platzhalter <the> abgebildet. Diese Generalisierung beinhaltet in dem bisherigen Modell allerdings nur den Nominativ für das Maskulinum und Femininum. Der dazugehörige Automat generiert lediglich das leere Symbol.

Für die Gebärdensprache ist ein vergleichbares Modell verwendet worden. Auf die Problematik der Namengebärden ist bereits im Kapitel 5.1 eingegangen worden. Bei der Modellierung wurde vereinfacht davon ausgegangen, dass ein technisches System zu der erkannten Namengebärde den amtlichen Namen der Person in Glossen (Großbuchstaben) zuordnet. Unterschiede zwischen beiden Sprachformen sind hier durch die zusätzlichen Gebärden WER und SELBST gegeben. Da sich diese auf ein Subjekt beziehen, sind die Glossen auch Bestandteil des Objekts *person*. Ergänzende sprachlichen Einheiten in dem DGS-Platzhalter <name> sind nicht verwendet worden, da hierzu leider keine genaueren Informationen vorliegen.

Modellierungsergebnis

Durch das Einsetzen der erstellten Platzhalter in den Transduktor zur *Elementargrammatik* liegt für beide Sprachformen jeweils ein erweiterter *Äußerungs-Bedeutungs-Transduktor* vor. Dieser wird für die praktischen Untersuchungen zur semantischen *Interpretation* und auch für die *Artikulation* verwendet.

Durch eine Ersetzung von Ausgabesymbolen durch die dazugehörigen Platzhaltersymbole können redundante Pfade im Modellierungsergebnis entstehen. Diese werden durch mehrfach vorkommende semantische Merkmale innerhalb einer Sprachäußerung verursacht. In dem untersuchten Beispiel werden bei jeder Sprachäußerung zwei Aktionen mit unterschiedlichen Werten benannt. In vielen Fällen führt eine Vertauschung der Aktionen wieder zu einer akzeptierten Wortfolge. So sind z. B. „*Peter steht und wartet dort*“ sowie „*Peter wartet und steht dort*“ jeweils Bestandteil der *Elementargrammatik*. In dem modellierten Beispiel beinhaltet der Automat für die Lautsprache vor dem Einsetzen der Platzhaltertransduktoren die folgenden 8 unterschiedlichen Sequenzen:

1. <loc_rel> <action> <name> und <action>
2. <loc_rel> <action> <the> <action> <name>
3. <loc_rel> <action> und <action> <name>
4. <name> <action> <loc_rel> und <action>
5. <name> <action> <action> <loc_rel>
6. <name> <action> und <action> <loc_rel>
7. <action> <action> <name> <loc_rel>
8. <the> <action> <name> <action> <loc_rel>

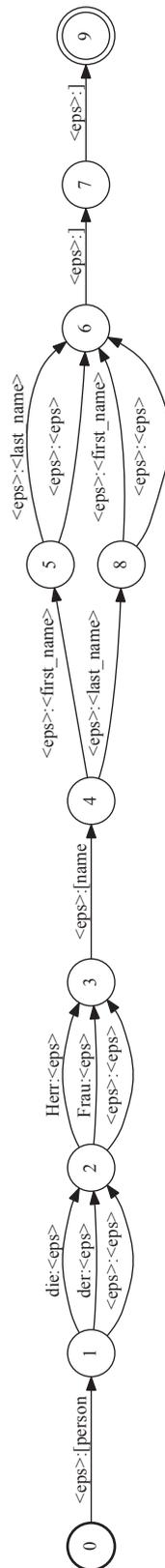


Abb. 5.12: Initialer FST für den Platzhalter „name“ (Lautsprache)

Für die Gebärdensprache ergeben sich hingegen aus den 10 verwendeten Sprachäußerungen 5 verschiedene Folgen mit Platzhaltersymbolen:

1. <name> <action> <action> <loc_rel>
2. <name> <loc_rel> <action> <action>
3. <loc_rel> <name> <action> <action>
4. <loc_rel> <name> <action> SELBST <action>
5. <loc_rel> <action> <action> <name>.

Eine Besonderheit stellt die Glossenfolge 6 der im Anhang A.3 aufgeführten Notationen dar. Die zwischen den beiden Handlungen artikuliert Glosse SELBST wurde nicht durch einen Platzhalter ersetzt. Dieser Aspekt ist durch die Platzhaltersymbolfolge 4 wiedergegeben. Die Glosse SELBST bezieht sich auf eine Person. Sie ist deshalb in den anderen Fällen Bestandteil des Platzhalters *name*.

5.3.4 Bewertung und Diskussion des Modellierungsergebnisses

Anhand eines Beispiels wurde eine Möglichkeit zur Konstruktion einer *mikrolokalen Grammatik* beschrieben. Dieses Prinzip kann auf andere *semantische Schemata* analog angewendet werden.

Untersucht wurde dabei die Erstellung und Verwendung universell einsetzbarer Platzhaltertransduktoren in *mikrolokalen Grammatiken*. Ein wesentlicher Vorteil besteht in der Wiederverwendbarkeit dieser Module. Einmal erstellte abstrakte Platzhalter können in unveränderter Form auch in anderen *Äußerungs-Bedeutungs-Transduktoren* eingesetzt werden.

Es ist zu beachten, dass Platzhaltertransduktoren eingangsseitig immer nur aufeinanderfolgende Sprachzeichen berücksichtigen können. In den Beispieläußerungen treten jedoch Phrasen, wie „*der wartende Peter*“ auf. Solche syntaktischen Varianten können auch nicht durch die Berücksichtigung möglicher Artikel im Platzhalter *name* abgedeckt werden. Soll nun die Wortfolge „*die wartende Sabine*“ interpretiert werden, so wird diese Sequenz nicht akzeptiert. Aus diesem Grund werden zusätzliche Platzhalter benötigt, welche Artikel und Pronomen in der *Elementargrammatik* ersetzen und das leere Symbol generieren.

Eine Substitution kann automatisch nach dem Einsetzen semantisch relevanter Platzhaltersymbole erfolgen. Damit wird eine Berücksichtigung des grammatischen Geschlechts zu variablen Nomen in der Sprachäußerung sichergestellt. Diese Funktion erfüllt in der Beispielgrammatik der Automat zu dem Platzhaltersymbol <*the*>.

Bei der Einteilung der semantischen Einheiten ist zumeist nicht sinnvoll, zwischen dem grammatischen und somit auch dem natürlichen Geschlecht zu unterscheiden. Trotzdem wäre eine Berücksichtigung dieser Differenzierung bei der Erstellung des Transduktors zur *mikrolokalen Grammatik* möglich.

Ein Ansatz dafür ist in [Rit13] und [RW14] dargestellt. Hierbei werden Ersetzungstabellen verwendet, welche das grammatische Geschlecht für Artikel, Pronomen und Gegenständen sowie das Genus von Lebewesen berücksichtigt. *Äußerungs-Bedeutungs-Transduktoren*, welche mit Hilfe dieser Ersetzungstabellen erzeugt werden, akzeptieren nur grammatisch richtige Sprachäußerungen. Die Funktion dieser Ersetzung kann im Prinzip auch durch differenzierte Platzhalterautomaten erreicht werden. Der Platzhalter für den Namen einer Person kann z. B. in weibliche und männliche Personen unterteilt werden. Eine mögliche Unterteilung in die Transduktoren *name_m* und *name_f* ist durch die Abb. B.4 im Anhang B gegeben. Die beiden Automaten unterscheiden sich durch die dazugehörigen geschlechtsbestimmenden Wörter *der/die* sowie *Herr/Frau* und die geschlechtsspezifischen Platzhaltersymbole. In diesem Beispiel beinhaltet der Platzhalter *first_name_m* nur männliche Vornamen und der Platzhalter *first_name_f* die aus einer Datenbasis extrahierten weiblichen Namen. Lassen sich Vornamen beiden Geschlechtern zuordnen, wie z. B. *Andrea* oder *Kim*, sind diese als Bestandteil beider Platzhalter einzusetzen. Zusätzlich ist neben der Modifikation der einzelnen Platzhalter der komplette durchgehende Pfad der mikrolokalen Grammatik zu betrachten. So sind beispielsweise in der Platzhaltersymbolfolge „*<loc_rel> <action> <the> <action> <name>*“ die zwei getrennten Folgen

1. *<loc_rel> <action> <the_f> <action> <name_f>* und
2. *<loc_rel> <action> <the_m> <action> <name_m>*

zu unterscheiden. Dadurch wird erreicht, dass der alleinstehende bestimmte Artikel in Numerus und Genus mit dem Subjekt übereinstimmt. Auch in der Gebärdensprache können Determinative artikuliert werden. In diesem Beispiel wäre durch die Verwendung eines Determinativs eine mögliche Verortung der Person „Peter“ im Gebärdenraum ausdrückbar. In dem gegebenen Kontext war dies jedoch nicht notwendig.

Die Erstellung eines *Äußerungs-Bedeutungs-Transduktors* zur *mikrolokalen Grammatik*, welcher nur grammatisch stimmige Sprachäußerungen akzeptiert, ist mit Mehraufwand realisierbar.

In bestimmten Fällen kann auch eine unüberwachte Adaption solcher Transduktoren notwendig sein. Dies ist z. B. dann wichtig, wenn domänenspezifisches Wissen zeitvariant ist. So müssen möglicherweise neue Datensätze, wie z. B. Einträge in einem Adressbuch oder neue Medienobjekte in einer Filmdatenbank, in den *Äußerungs-Bedeutungs-Transduktoren* berücksichtigt werden. Dazu ist lediglich eine Aktualisierung der Platzhaltertransduktoren nötig. Allerdings ist nicht immer sichergestellt, dass auch die erforderlichen grammatischen Informationen zur Verfügung stehen.

Ein weiterer Nachteil dieses Modellierungsansatzes besteht in der erhöhten Abweisungs-

wahrscheinlichkeit von Benutzeräußerungen in sprachverarbeitenden Systemen. Eine weniger restriktive *mikrolokale Grammatik* hat in der Praxis den Vorteil, dass der ihr zugeordnete Transduktor eine weitaus größere Anzahl an Sprachäußerungen akzeptiert, welche jedoch trotzdem in das dazugehörige *semantische Schema* passen. Problematisch ist der Modellierungsansatz hingegen, wenn die erstellten *Äußerungs-Bedeutungs-Transduktoren* auch für die Artikulation eingesetzt werden. Eine stärkere Abstraktion bei der Modellierung führt letztendlich dazu, dass durch die Transduktoren linguistische Informationen nicht exakt spezifiziert sind. Dadurch steigt bei der Artikulation die Anzahl generierbarer Wortfolgen stark an. Ein großer Teil dieser Sprachzeichenfolgen ist allerdings grammatisch unstimmig. Alternativ können die Artikulationsgrammatiken für ein Sprachdialogsystem auch separat modelliert werden.

Kapitel 6

Ergebnisse der semantischen Interpretation und Artikulation für das modellierte Schema

Die modellierten *Äußerungs-Bedeutungs-Transduktoren* zu dem in Kapitel 5 aufgeführten *semantischen Schema* werden in einer Beispielanwendung genutzt. Eine vom Anwender vorgegebene Sprachzeichenfolge wird hierbei semantisch interpretiert. Wahlweise kann aus der dazu erstellten Bedeutungsrepräsentation wieder eine Wortsequenz der deutschen Lautsprache oder eine Glossenfolge der Deutschen Gebärdensprache generiert werden.

Zusätzlich sind endliche Transduktoren erstellt worden, welche den im Kapitel 3.1.3 beschriebenen Interpretationsprozess für dasselbe *semantische Schema* realisieren. Dabei kommt u. a. ein „mikrolokales“ *konzeptuelles Sprachmodell* für die deutsche Lautsprache zum Einsatz. Dadurch ist ein direkter Vergleich des Modellierungsvorgangs und der erzielten Ergebnisse möglich. Alle Automatenoperationen wurden mittels *OpenFst* [ARS+07] realisiert.

6.1 Semantische Interpretation

Für die automatenbasierte semantische Analyse von sprachlichem Verhalten sind geeignete Modelle notwendig. Im Abschnitt 5.3.3 wurde anhand der Beispielanwendung exemplarisch die Modellierung eines *Äußerungs-Bedeutungs-Transduktors* T_{UM} für ein

konkretes semantisches Schema S beschrieben. Diese Transduktoren generieren zu einer akzeptierten Sprachäußerung eine dazugehörige *MWR-Zeichenkette*.

Erfolgt durch Transduktoren eine phrasenweise Zuordnung von semantischen Merkmalen, so kann mit einer Phrase u. U. eine Vielzahl möglicher semantischer Kategorien assoziiert werden. Wie bereits in Kapitel 3.1.3 beschrieben, ist aus diesem Grund die Verwendung eines *konzeptuellen Sprachmodells* für die semantische Dekodierung sinnvoll. Als Ergebnis dieses Interpretationsvorgangs erhält man eine Folge gewichteter Merkmal-Wert-Paare.

Für eine vergleichende Untersuchung soll die Menge von Sprachäußerungen $\mathcal{U}(S)$ zu dem semantischen Schema S auch durch dieses Verfahren semantisch analysiert werden können. Ausgangspunkt ist jeweils eine durch den Anwender vorgegebene Wortfolge einer Sprachäußerung. Zu dieser wird der Transduktor T_W erzeugt, welcher genau die ausgewählte Sprachzeichenfolge generiert. Grundsätzlich kann natürlich auch ein gewichteter Worthypothesengraph mit mehreren Erkennalternativen verwendet werden. Als Beispiel dient nachfolgend die Sprachäußerung \mathbf{x} = „Wartend steht Peter dort drüben“. Diese wird sowohl von der *Elementargrammatik* als auch von der *mikrolokalen Grammatik* akzeptiert ($\mathbf{x} \in \mathcal{U}(S) \cap \mathcal{U}(M)$). In den Einträgen der Lexika zu den eingesetzten Grammatiken sind Umlaute und der Buchstabe ß ersetzt worden. Zudem beginnen nur die Substantive mit einem Großbuchstaben. Aus diesem Grund müssen die zu analysierenden Wortfolgen entsprechend konvertiert werden. Dazu gehört auch die Entfernung von eventuell vorkommenden Satzzeichen.

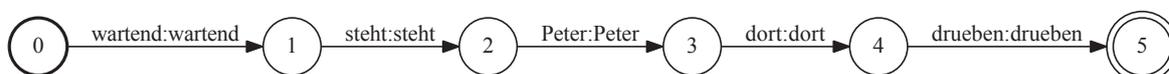


Abb. 6.1: Transduktor T_W für die ausgewählte Wortfolge

6.1.1 Interpretation mit Äußerungs-Bedeutungs-Transduktor

Das Kompositionsergebnis aus Worthypothesengraph T_W und *Äußerungs-Bedeutungs-Transduktor* T_{UM} ist der Transduktor T_{MWR} . Dieser enthält für die interpretierte Beispieläußerung \mathbf{x} = „Wartend steht Peter dort drüben.“ zwei durchgehende Pfade. Diese unterscheiden sich lediglich durch die beiden möglichen Deutungen des Wortes „Peter“. Zum einem kann der *semantische Anker* „[person[name[first_name[Peter]]]]“ interpretiert

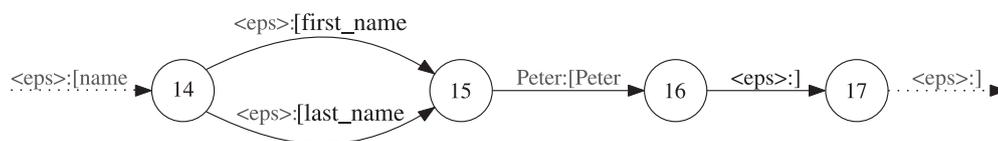


Abb. 6.2: Interpretationsergebnis - Ausschnitt des Transduktorgraphen T_{MWR}

werden und zum anderen wird durch den *Äußerungs-Bedeutung-Transduktor* T_{UM} auch das *Konzept* „[person[name[last_name[Peter]]]]“ der Sprachzeichenfolge \mathbf{x} zugeordnet. Der zu diesem Aspekt relevante Bereich des Automatengraphen wird in Abb. 6.2 gezeigt. Das Interpretationsergebnis ist in diesem Fall nicht eindeutig. In der Beispielanwendung wird aus dem Transduktor T_{MWR} nun zufällig ein durchgehender Pfad ausgewählt. In diesem Fall ist die Wahrscheinlichkeit groß, dass es sich bei „Peter“ um einen Vornamen handelt. Bei der Verwendung gewichteter Worthypothesengraphen und stochastischen semantischen Grammatiken ist hingegen der durchgehende Pfad mit den geringsten Gesamtkosten auszuwählen. Nach Entfernung der leeren Symbole $\langle \text{eps} \rangle$ in den Ausgabezeichenfolgen des Transduktors T_{MWR} erhält man die *MWR-Zeichenkette*. Die Sequenz:

$$FVR[PETER[action/wait]][action/stand]][person[name[first_name/Peter]]][loc/rel/there]]]$$

ist eine der beiden möglichen semantischen Zeichenfolgen des interpretierten Beispielsatzes \mathbf{x} . Diese wird im Anschluss geparkt und in eine *Merkmal-Werte-Relation* überführt. Wie in Kapitel 4.3 erläutert, erfolgt dabei eine Kaskadierung gleichnamiger Geschwisterknoten. Abb. 6.3 zeigt die als Baumstruktur dargestellte *Merkmal-Werte-Relation*, welche aus der *MWR-Zeichenfolge* erzeugt wurde.

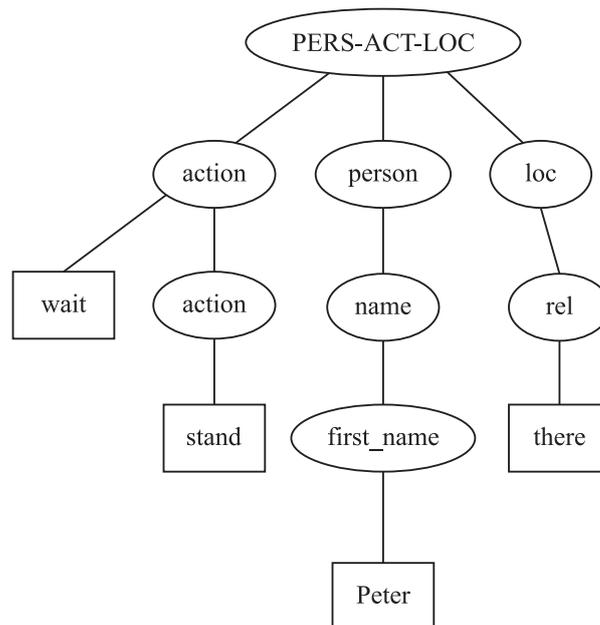


Abb. 6.3: Merkmal-Werte-Relation als Ergebnis des Interpretationsprozesses

6.1.2 Interpretation mit konzeptuellem Sprachmodell

Für das beschriebene *semantische Schema* der Beispielanwendung wird die Interpretation von Sprachäußerungen nach der im Kapitel 3.1.3 beschriebenen Vorgehensweise auf

Grundlage von [RR07, Din10, HDR⁺11] durchgeführt. Eine Erstellung der benötigten Transduktoren erfolgt allerdings nur für die deutsche Lautsprache. Das *konzeptuelle Sprachmodell* ist dabei kein N-Gramm-Modell für semantische *Konzeptfolgen*, da es Wörter bzw. Generalisierungen anstatt Datenwerte berücksichtigt. Aus diesem Grund muss in einem multilingualen Sprachdialogsystem für jede zu berücksichtigende natürliche Eingabesprache ein individuelles stochastisches Folgenmodell erstellt werden. Nach Gleichung 3.18 wird die semantische Analyse durch die Komposition von 5 Transduktoren ($T_W, T_G, T_{G2K}, T_{KSM}$ und T_{MWE}) erreicht. Ausgangspunkt ist auch hier der ungewichtete Transduktor T_W (vgl. Abb. 6.1), welcher den Worthypothesengraphen repräsentiert. Dieser akzeptiert und generiert gleichzeitig auch mögliche Wortfolgen der analysierten Sprachäußerung. Die vom gewichteten Transduktor T_{SA} erzeugten Folgen von Attribut-Wert-Paaren stellen das Interpretationsergebnis dar. Dabei kann die Syntax der Ausgabesymbolfolge durch den erstellten Transduktor T_{MWE} zur *Merkmal-Wert-Extraktion* beeinflusst werden.

Beim Interpretationsprozess werden bestimmte Wörter durch ihre Klassenbezeichnung ersetzt. Diese Aufgabe wird durch den *Generalisierungstransduktor* T_G realisiert. Dieser substituiert in der Beispielanwendung u. a. bestimmte Artikel (*BEST_ARTIKEL*), Personennamen (*NAME*) und die Wörter, welche die Aktionen ausdrücken (*STEHEN, WARTEN, SITZEN, RAUCHEN, LIEGEN*) mit der Klassenbezeichnung. Diese ist hier in runden Klammern bereits angegeben. Zur Unterscheidung von nicht ersetzten Wörtern werden für die Kategorienamen Großbuchstaben verwendet. Dabei können durchaus Parallelen zu den in Kapitel 5.3.3 modellierten Platzhaltern festgestellt werden. So beinhalten die Kategorien für die Handlungen bereits alle möglichen Verbflexionen. Zusätzlich können auch Synonyme durch eine gemeinsame Kategoriebezeichnung zusammengefasst werden. Die Mächtigkeit dieses verallgemeinernden Modellierungsansatzes kann u. U. auch problematisch sein, da mögliche lexikalische Bedeutungen der Sprachzeichen einer Klasse nicht immer einheitlich sind.

Durch die Komposition mit einem Automaten T_{G2K} erfolgt eine Zuordnung der potenziellen Konzeptmarkierungen zu den Sprachzeichen. Für die Identifikation der möglichen Konzeptbeschriftungen wird die semantisch annotierte Lernstichprobe genutzt. Zum Vergleich ist ein Ausschnitt des Automatengraphen zum Transduktor T_{G2K} in Abb. B.6 des Anhangs B dargestellt. Durch diesen Kompositionsschritt werden beispielsweise der Wortklasse *NAME* vier mögliche Konzeptmarkierungen zugeordnet, da in dieser Klasse Vornamen und Familiennamen enthalten sind. Dieser Zusammenhang ist in Abb. 6.4 dargestellt. Eine feinere Einteilung von Klassen wäre in diesem Fall jedoch sinnvoll. So könnte man für Vor- und Nachnamen z. B. die separaten Kategorien *NAME.FIRST_NAME* und *NAME.LAST_NAME* verwenden. Die Problematik, dass auch Wörter, wie z. B. der Personennamen „Peter“, zu mehreren Generalisierungen zugeordnet werden, wird durch diese Vorgehensweise nicht gelöst. Eine Komposition mit dem

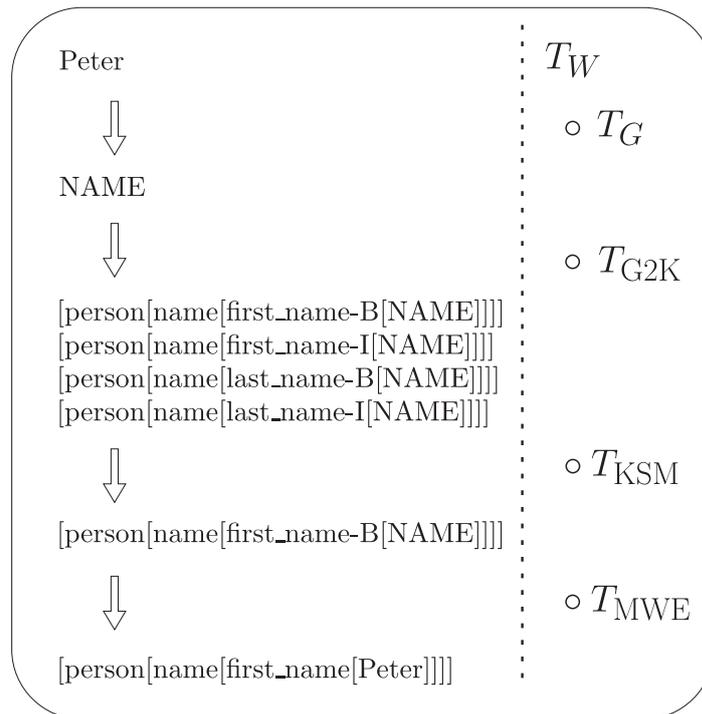


Abb. 6.4: Darstellung der Übersetzungsprozesse einzelner Transduktoren am Beispiel des Sprachzeichens „Peter“. Dabei erfolgt nach der Komposition mit dem Sprachmodell T_{KSM} die Auswahl der wahrscheinlichsten Hypothese.

konzeptuellen Sprachmodell T_{KSM} liefert einen gewichteten Hypothesgraphen. Abb. 6.5 zeigt den relevanten Teil des Kompositionsergebnisses zu der analysierte Wortfolge „Wartend steht Peter dort drüben.“. Dieses Zwischenergebnis enthält unterschiedliche Konzeptmarkierungen zu dem Sprachzeichen „Peter“. Diesen sind jedoch abweichende Kantengewichte zugeordnet. Der zur Bewertung verwendete endliche Transduktor T_{KSM} repräsentiert hier ein geglättetes „mikrolokales“ Bigramm-Sprachmodell für semantisch annotierte Sprachzeichenfolgen. Weitere Ausführungen zur Erzeugung dieses Bigramm-Sprachmodells werden in Kapitel 6.3 gemacht.

Bei der Erstellung des Trainingsmaterials sind einige Wörter durch die Klassenbezeichnung zu ersetzen. Zusätzlich müssen die Wörter/Generalisierungen mit Konzept-



Abb. 6.5: Ausschnitt des Transduktors $T_W \circ T_G \circ T_{G2K} \circ T_{KSM}$ zur Beispielaussage „Wartend steht Peter dort drüben.“

markierungen versehen werden. Diese enthalten hierbei immer die Markierungen für Beginning (B) und Inside (I) (vgl. Kapitel 3.1.3. So wird hier z. B. dem Wort „steht“ die Notation [action-B[STEHEN]] zugeordnet. Für die Erstellung des Folgenmodells wurden insgesamt 225 Wortsequenzen verwendet. Diese stammen aus der Menge von Sprachäußerungen, welche in den Anhängen A.1 und A.2 aufgelistet sind. Zusätzlich wurden 20 Konzeptmarkierungsfolgen zu frei ausgedachten Wortfolgen in das Trainingsmaterial aufgenommen. Letztendlich gehören zu dem „mikrolokalen“ *konzeptuellen Sprachmodell* 1929 Wörter (Tokens) und 36 Lexikoneinträge. Die geringe Anzahl an Lexikoneinträgen resultiert dabei aus der umfangreichen Verwendung von Wortklassen.

In der Beispielanwendung wird nur der durchgehende Pfad mit den geringsten Kosten $w(\pi_i)$ weiterverarbeitet. Zu diesem erfolgt eine regelbasierte Bestimmung von semantischen *Konzepten* durch den zu konstruierenden Transduktor T_{MWE} . In diesem Fall wurde für jedes Merkmal ein Transduktor T_{M_i} erstellt, welcher eine semantisch annotierte Phrase in das dazugehörige *Konzept* übersetzt. Die Erstellung des Automaten T_{MWE} erfolgt somit durch

$$T_{MWE} = \left(\bigoplus_{i=1}^N T_{M_i} \right)^* . \quad (6.1)$$

Der *Kleenesche Abschluss* erlaubt die Merkmal-Wert-Extraktion beliebiger Sequenzen. Das Merkmal ergibt sich direkt aus der Konzeptmarkierungsbezeichnung. Jedem detektierten Attribut wird ein individueller normierter Datenwert zugeordnet. Sprachzeichen, für die keine konkrete Bedeutungszuordnung erfolgte, werden in das „Dummy-Konzept“ [null] übersetzt. Für die interpretierte Wortfolge „*Wartend steht Peter dort drüben.*“ erhält man letztendlich die Symbolsequenz:

```
[action[wait : 3.69]][action[stand : 2.65]][person[name[first_name[Peter : 1.58]]]
[loc[rel[there : 2.58]]]
```

Zur Notation der *Konzepte* ist in diesem Fall eine Klammerstruktur verwendet worden.

6.1.3 Vergleich der Interpretationsvorgänge

Für die semantische Analyse eines erkannten sprachlichen Ausdrucks wurden zwei verschiedene Ansätze exemplarisch umgesetzt. In beiden Fällen werden endliche Automaten verwendet. Dies ermöglicht auch einen direkten Vergleich der beiden Verfahren.

Zum einen wurde die Interpretation durch den für das *semantische Schema* erstellten *Äußerungs-Bedeutungs-Transduktor* (vgl. Kapitel 5) rein regelbasiert durchgeführt. Dabei stellten die Akquise von Sprachäußerungen zur *Elementargrammatik* und die manuelle Modellierung der Automaten für die Platzhalter den größten Aufwand dar. Prinzipiell können Sprachäußerungen sehr kurze Wortfolgen sein oder aber auch mehrere Sätze beinhalten. Schon allein die in dieser Arbeit erstellten *Elementargrammatiken* zeigen, dass viele syntaktische Variationen möglich sind. Wie im Kapitel 4 beschrieben,

ist mit Hilfe einer geeigneten Lernstichprobe auch die Erstellung einer stochastischen *mikrolokalen Grammatik* möglich. Als Ergebnis der semantischen Interpretation erhält man *beschriftete partielle Ordnungen*.

Durch die Verwendung von Platzhaltern im *Äußerungs-Bedeutungs-Transduktor* zur *mikrolokalen Grammatik* besteht die Möglichkeit, dass mehrere Bedeutungshypothesen erzeugt werden. Dies ist dann der Fall, wenn einem Sprachzeichen mehrere *semantische Anker* zugeordnet werden können. In der Beispielanwendung trifft dies z. B. auf das Sprachzeichen *Peter* zu, welches ein Vor- und auch ein Nachname sein kann. In vielen Fällen liefert ein *Äußerungs-Bedeutungs-Transduktor* bei der semantischen Analyse vermutlich ein eindeutiges Resultat.

Weiterhin wurde durch die Komposition gleich mehrerer Transduktoren ein statistischer Ansatz für das Sprachverstehen verwendet. Hierbei kommt ein *konzeptuelles Sprachmodell* zum Einsatz. Damit kann für die Dekodierung auch die geschätzte Verbundwahrscheinlichkeit von generalisierter Wortfolge und Konzeptmarkierungssequenz berücksichtigt werden. Bestehende Relationen zwischen *Attribut-Wert-Paaren* können mit diesem Ansatz allerdings nicht direkt hergestellt werden. Die dafür nötigen Informationen sind nicht in den Transduktoren enthalten. Hierfür wird zusätzliches domänen- und situationsspezifisches Wissen benötigt. Eine hierarchische Strukturierung wäre z. B. nachträglich durch ein *slot-filling* unter Verwendung geeigneter dialogaktabhängiger semantischer Schablonen möglich.

Analog zu den *Äußerungs-Bedeutungs-Transduktoren* werden durch die hier erstellten Modelle ebenfalls Klammerstrukturen generiert. Diese Klammerausdrücke entsprechen der festgelegten Syntax von *MWR-Zeichenketten*. Auf diese Weise lässt sich die Bedeutungsstruktur in Form eines gerichteten azyklischen Graphen darstellen. Für die regelbasierte Bestimmung der *Attribut-Wert-Paare* aus den semantisch annotierten Sprachzeichenfolgen ist auch hier ein entsprechender zusätzlicher manueller Modellierungsaufwand erforderlich.

Aus der Interpretation der Folge „*[action[wait : 3.69]][action[stand : 2.65]][person[name[first_name[Peter : 1.58]]][loc[rel[there : 2.58]]]*“ resultiert die in Abb. 6.6 dargestellte Baumstruktur. Somit erhält man auf verschiedenen Wegen ein ähnliches Interpretationsergebnis zu demselben ungewichteten Transduktor T_W . Im Vergleich zur MWR in Abb. 6.3 sind in dieser Bedeutungsstruktur die Datenwerte aufgrund der Sprachmodellbewertung gewichtet. Zu beachten ist hierbei, dass ein *konzeptuelles Sprachmodell* in der Regel aus dem gesamten Korpus erstellt wird. Aus diesem Grund enthält die erzeugte Bedeutungsstruktur auch keine übergeordnete handlungsspezifische Bezeichnung. Entsprechend enthält der Wurzelknoten in Abb. 6.6 keine spezifische Beschriftung. Sowohl das Sprachmodell als auch die Modelle zur *Attribut-Wert-Extraktion* werden dabei unabhängig vom Interaktionskontext eingesetzt. Abhängigkeiten zwi-

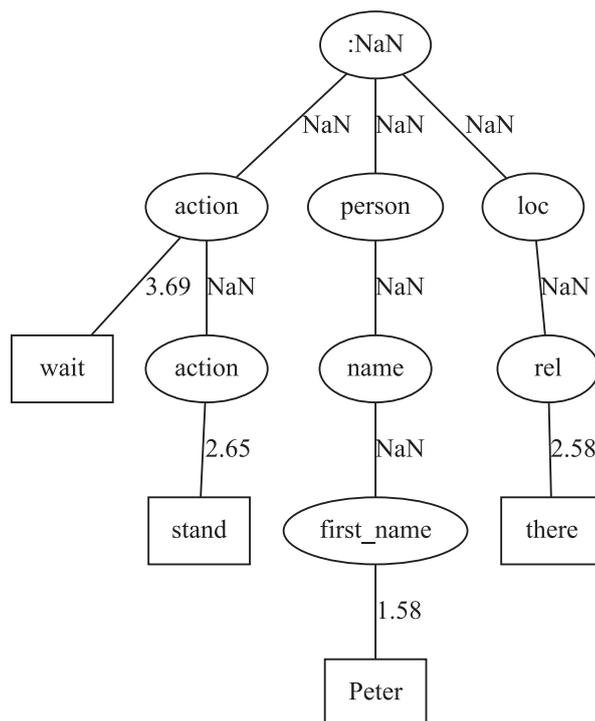


Abb. 6.6: Interpretationsergebnis als gewichtete Merkmal-Werte-Relation

schen semantischen Einheiten innerhalb längerer Sprachäußerungen werden mit einem *konzeptuellen Sprachmodell* nicht ausreichend berücksichtigt.

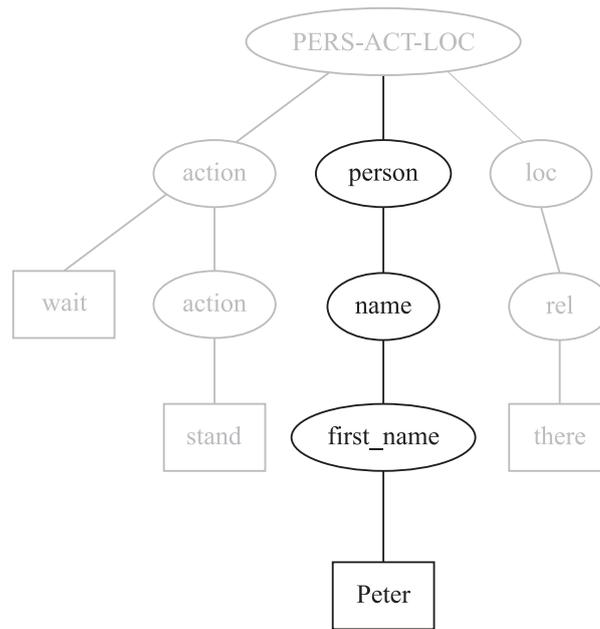
6.2 Artikulation einer Bedeutungsstruktur mit dem inversen Äußerungs-Bedeutungs-Transduktor

Ausgangspunkt der Generierung einer Wortfolge ist die *Inhaltsfestlegung*. Für die praktischen Untersuchungen entspricht die zu artikulierende Bedeutungsstruktur allerdings dem Ergebnis des Interpretationsprozesses. Wie im Abschnitt 4.4 beschrieben, ist aus der gegebenen *Merkmal-Werte-Relation* zunächst ein Automat T_{MWP} mit semantischen Symbolfolgen zu erstellen. Jeder durchgehende Pfad des Transduktors entspricht dabei einer gültigen *MWR-Zeichenkette*. In Abb. 6.7 ist der zu einem *semantischen Anker* zugehörige Subgraph innerhalb der *Merkmal-Werte-Relation* hervorgehoben. Die Wurzel des gerichteten Baumes enthält hierbei die konkrete Bezeichnung des *semantischen Schemas*. In Sprachdialogsystemen kann der verwendete Begriff z. B. den Typ des Dialogaktes repräsentieren. Der *Linearisierungsprozess* liefert für die in Abb. 6.3 gezeigte *Merkmal-Werte-Relation* folgende Sequenzen s_i :

```

[action [wait ] ] [action [stand ] ] [person [name [first_name [Peter ] ] ] ] [loc [rel [there ] ] ]
[person [name [first_name [Peter ] ] ] ] [action [wait ] ] [action [stand ] ] [loc [rel [there ] ] ]

```



[person[name[first_name[Peter]]]]

Abb. 6.7: Zusammenhang zwischen semantischem Anker in einer Merkmal-Werte-Relation und der dazugehörigen Symbolfolge aus der MWR-Zeichenkette

[action [wait]] [person [name [first_name [Peter]]]] [action [stand]] [loc [rel [there]]]
 [loc [rel [there]]] [action [wait]] [action [stand]] [person [name [first_name [Peter]]]]
 [action [wait]] [loc [rel [there]]] [action [stand]] [person [name [first_name [Peter]]]]
 [action [wait]] [action [stand]] [loc [rel [there]]] [person [name [first_name [Peter]]]]
 [person [name [first_name [Peter]]]] [loc [rel [there]]] [action [wait]] [action [stand]]
 [loc [rel [there]]] [person [name [first_name [Peter]]]] [action [wait]] [action [stand]]
 [action [wait]] [person [name [first_name [Peter]]]] [loc [rel [there]]] [action [stand]]
 [person [name [first_name [Peter]]]] [action [wait]] [loc [rel [there]]] [action [stand]]
 [loc [rel [there]]] [action [wait]] [person [name [first_name [Peter]]]] [action [stand]]
 [action [wait]] [loc [rel [there]]] [person [name [first_name [Peter]]]] [action [stand]]

Die Zeichenketten sind aus Gründen der Übersichtlichkeit verkürzt dargestellt. Jede Symbolfolge s_i muss tatsächlich noch in die Zeichenfolge „FVR[PERS-ACT-LOC s_i]“ eingesetzt werden. Auf Grundlage dieser 12 möglichen Sequenzen zu der vorgegebenen *beschrifteten partiellen Ordnung* wird der Transduktor T_{MWP} erzeugt. Im konkreten Fall akzeptiert und generiert dieser die segmentierten Folgen der *MWR-Zeichenketten*. Die Abb. B.5 des Anhangs B zeigt den zugehörigen Automatengraphen zu den oben aufgeführten Symbolfolgen. Diese 12 semantischen Zeichenketten werden anschließend durch den Transduktor zur *inversen mikrolokalen Grammatik* analysiert.

Ergebnisse der Textgenerierung mit inversen Äußerungs-Bedeutungs-Transduktoren

Durch die Komposition mit *inversen Äußerungs-Bedeutungs-Transduktoren* erhält man den Automaten T_{AW} . Dieser generiert zu den akzeptierten *MWR-Zeichenketten* die dazugehörigen Wort- bzw. Glossenfolgen der Zielsprache. Die Ausgabesymbolfolge zu einem durchgehenden Pfad dieses Automaten ist einer artikulierten Sprachäußerung zugeordnet. Vor der Anzeige des Artikulationsergebnisses ist noch eine Formatierung der Ausgabezeichenfolge vorzunehmen. Hierbei erfolgt die Entfernung des Startsymbols, eine Substituierung vom Umlauten, das Einsetzen von Satzzeichen und die Großschreibung am Satzanfang.

Zu der Bedeutungsrepräsentation der interpretierten Wortfolge \mathbf{x} „*Wartend steht Peter dort drüben.*“ werden durch den *inversen Äußerungs-Bedeutungs-Transduktor* insgesamt 57024 Sprachzeichensequenzen der deutschen Lautsprache generiert. Für die Deutsche Gebärdensprache erzeugt der Sprachgenerierungsprozess zu derselben MWR lediglich 30 Glossenfolgen. Diese sind im Anhang C.1 aufgelistet. Ursachen dafür sind die detailliertere Modellierung der Platzhalter in der Lautsprachgrammatik und die durchschnittlich höhere Anzahl an Sprachzeichen in den Äußerungen der deutschen Lautsprache. Durch die kombinatorische Explosion beim durchgeführten Sprachgenerierungsprozess entsteht also u. U. eine Vielzahl unterschiedlicher Wortfolgen. Viele dieser erzeugten Wortsequenzen sind jedoch grammatisch unstimmg. Auch die Syntax der Glossenfolge ist nicht immer korrekt. Die Fragedeterminative *WAS* und *WER* treten in den generierten Sequenzen auch am Satzanfang auf. Dies ist für die DGS eher ungewöhnlich. Diese syntaktischen Strukturen kamen auch nicht in den Beispielsätzen vor. Um eine ausreichende Akzeptanz des technischen Systems zu erreichen ist es jedoch sinnvoll, dass die erzeugte Systemreaktion der Grammatik der verwendeten natürlichen Sprache \mathcal{L} entspricht. Möglicherweise sind die erzeugten Sprachäußerungen für den Menschen jedoch trotzdem verständlich. Auf eine Evaluierung der *Verständlichkeit* und der *Kommunizierbarkeit* dieser erzeugten Sprachäußerungen wurde aus Zeitgründen verzichtet. Ansätze und Möglichkeiten dafür findet man u. a. in [Möl10].

6.3 Wortfolgenbewertung durch ein Sprachmodell

Bei der Erzeugung der möglichen Wortfolgen werden alle textuellen Korrelate zu den *Konzepten* der Bedeutungsrepräsentation berücksichtigt. Durch die universelle Modellierung der Platzhalter in dem *Äußerungs-Bedeutungs-Transduktor* erfolgt bei dem Sprachgenerierungsprozess eine Übergenerierung von Sprachzeichenfolgen. Ein Großteil dieser kombinatorisch möglichen Wortsequenzen sind jedoch grammatisch

unstimmig. Die Verwendung eines Sprachmodells der Zielsprache führt hierbei zu einer Bewertung der Lösungsmenge. Zu diesem Zweck wurden in den eigenen Untersuchungen zwei unterschiedliche N -Gramm-Sprachmodelle verwendet. Ziel dieses Vorgangs ist die Bestimmung einer Rangfolge der generierten Sprachäußerungen.

6.3.1 Verwendung eines mikrolokalen Sprachmodells

Ein N -Gramm-Sprachmodell kann als eine stochastische reguläre Grammatik aufgefasst werden [MPR08]. Das Sprachmodell wird hier durch den gewichteten Transduktor T_{SM} repräsentiert. Die Automatenzustände stellen die Worthistorie dar. In dem erstellten Sprachmodell wird für die Zustandsübergänge e_i dasselbe Ein- und Ausgabesymbol ($x_i = y_i$) verwendet. Dabei ist aus den geschätzten Wahrscheinlichkeiten der Wortübergänge ein Kantengewicht berechnet worden. In dieser Arbeit wird ein Bigramm-Sprachmodell ($N = 2$) verwendet, welches lediglich eine Vorgeschichte der Länge 1 besitzt. Diese Vorgehensweise entspricht einer Markov-Kette 1. Ordnung. Die geschätzte Wahrscheinlichkeit $P^*(x_i|x_{i-1})$ zu dem untersuchten Wort x_i ergibt sich in diesem Fall direkt aus der relativen Bigrammhäufigkeit. Diese kann durch die Gleichung

$$P^*(x_i|x_{i-1}) = \frac{\#(x_{i-1}x_i)}{\sum_n \#(x_{i-1}x_n)} \quad (6.2)$$

bestimmt werden. Zur Normierung wird die absolute Häufigkeit aller Bigramme verwendet, welche die konkrete Wortvorgeschichte x_{i-1} besitzen. In dem eigenen Sprachmodell dient das Symbol $\langle \text{start} \rangle$ als Anfangsmarkierung einer Sprachäußerung. Dieses Startsymbol ist in die Sprachäußerungen des Trainingskorpus eingefügt worden und ist dementsprechend auch für jede zu bewertende Sprachzeichenfolge zu verwenden. Auf diesem Weg ist es möglich, auch dem ersten Wort einer Sprachsymbolsequenz eine geschätzte Wahrscheinlichkeit zuzuordnen. Auf die Einführung eines Endsymbols wurde hingegen verzichtet.

Für die hier verwendeten Sprachmodelle stand ein relativ kleiner Trainingsdatenumfang zur Verfügung. Bei der Sprachmodellerstellung wurde eine einfache Glättung (*add- α smoothing*) verwendet. Konkret wurde $\alpha = 1$ (*add-one smoothing*) gewählt. In diesem Fall wird die relative Häufigkeit der Bigramme durch

$$P^*(x_i|x_{i-1})_{Add-\alpha} = \frac{\#(x_{i-1}x_i) + \alpha}{(\sum_n \#(x_{i-1}x_n)) + \alpha V} \quad (6.3)$$

$$P^*(x_i|x_{i-1})_{Add-One} = \frac{\#(x_{i-1}x_i) + 1}{(\sum_n \#(x_{i-1}x_n)) + V}$$

bestimmt. Dabei entspricht V der Vokabulargröße, also der Anzahl von Lexikoneinträgen (*Types*) zu den Trainingsdaten. Bei der praktischen Umsetzung wurden jedoch

die Bigramme entfernt, bei der das Startsymbol an der 2. Stelle steht. Da in der Beispielanwendung keine Sprachäußerungen verkettet werden, stellen diese Bigramme auch unmögliche Ereignisse dar. Im Allgemeinen führt das *add-one smoothing* allerdings zu einer Überbewertung der ungesehenen Bigramme der Lernstichprobe [Fin14]. Für das „mikrolokale“ Sprachmodell der deutschen Lautsprache wurden 440 Sprachäußerungen erstellt. Diese resultieren vor allem aus den im Anhang A.1 und A.2 aufgelisteten Beispielsätzen. Dabei wurden u. a. die in den Platzhaltern verwendeten alternativen Aktionen und Personennamen berücksichtigt. 224 Wortfolgen des Korpus wurden relativ frei von Vorgaben erstellt. Bei diesen sind allerdings jeweils immer Wörter genutzt worden, die auch in der mikrolokalen Grammatik des lautsprachlichen *Äußerungs-Bedeutungs-Transduktors* enthalten sind. Der für das Bigramm-Sprachmodell erstellte Textkorpus besteht insgesamt aus 3361 Wörtern (Tokens) und $V = 343$ verschiedenen Lexikoneinträgen (Types). Die Kleinschreibung von Wörtern wurde bei einer Position am Satzanfang nicht geändert. Mit $P^*(x_1 = \langle \text{start} \rangle) = 1$ ergibt sich die geschätzte Wahrscheinlichkeitsbewertung einer Wortfolge \mathbf{x} durch das Bigramm-Modell mit

$$P^*(\mathbf{x}) = P^*(x_2|\langle \text{start} \rangle) \cdot P^*(x_3|x_2) \cdot P^*(x_4|x_3) \cdot P^*(x_5|x_4) \cdot \dots$$

$$P^*(\mathbf{x}) = \prod_{i=2}^L P^*(x_i|x_{i-1}). \quad (6.4)$$

Die Bewertung des Transduktors aus $T_{MWP} \circ T_{UM}^{-1}$ durch das Sprachmodell T_{SM} liefert als Ergebnis gewichtete Wortfolgen der Zielsprache. Das Pfadgewicht ergibt sich nach Gleichung 2.1 durch die \otimes -Operation der beteiligten Kantengewichte $w(e_i)$. Da durch die Multiplikation von Wahrscheinlichkeitswerten diese Gewichte sehr klein werden, ist es üblich, die Zahlenwerte zu logarithmieren. In diesem Fall erfolgt die Berechnung des Kantengewichts $w(e_i)$ durch

$$w(e_i) = -\log_e P^*(x_i|x_{i-1}) = -\ln(P^*(x_i|x_{i-1})). \quad (6.5)$$

Das Pfadgewicht $w[\pi]$ ergibt sich jetzt durch die Addition der logarithmischen Werte. Analog zu Gleichung 6.5 wird das Gesamtgewicht für die Wortfolge \mathbf{x} durch das Sprachmodell mit

$$T_{SM}(\mathbf{x}) = -\ln(P^*(\mathbf{x})) \quad (6.6)$$

zugeordnet. Zu dem Kompositionsergebnis $T_{AW} = T_{MWP} \circ T_{UM}^{-1} \circ T_{SM}$ kann man für alle durchgehende Pfade entsprechende Pfadgewichte $w[\pi]$ berechnen. Anfangs- und Schlusszustände sind hier mit keiner Gewichtungsfunktion belegt. Die Auswahl der wahrscheinlichsten Ausgabezeichenfolgen $\hat{\mathbf{y}}$ zu der Menge von durchgehenden Pfaden $\mathcal{U}(I, F)$ aus dem Übersetzungsprozess erfolgte zunächst durch die Bestimmung der minimalen Pfadkosten.

Hierfür erzeugt der *N-ShortestPath-Algorithmus* der OpenFst-Bibliothek aus dem Automaten T_{AW} einen Transduktor $T_{AW_{N-best}}$ mit einer vorgegeben Pfadanzahl N . Die

Bildung einer Rangfolge anhand des Pfadgewichts für einen Automaten kann durch

$$T_{AW_{N-best}} = \bigoplus_{n=1}^N \arg \text{rank}_n w[\pi] \quad (6.7)$$

ausgedrückt werden. Dieser Prozess liefert jedoch keine optimalen Ergebnisse, denn bei diesem Ansatz werden kürzere Wortfolgen (= kleinere Anzahl an gewichteten Zustandsübergängen) bevorzugt. Längere Wortfolgen der Zielsprache treten durch die Summierung der Kantengewichte in den N -besten-Listen seltener auf.

Gesucht wird also ein Verfahren für die Bestimmung einer Rangfolge, welches die Pfadlänge der generierten Sprachzeichenfolgen berücksichtigt.

6.3.2 Normierte Gesamtbewertung

Zur Normierung der Pfadkosten bietet sich zunächst eine informationstheoretische Betrachtung an. Ein diskreter stochastischer Prozess erzeugt Wortfolgen \mathbf{x} . Der mittlere Informationsgehalt (Entropie) einer Zufallsvariablen \mathbf{X} über alle möglichen Zeichenfolgen \mathbf{x} der endlichen Länge $|\mathbf{x}| = L$ ist durch

$$H(\mathbf{X}) = - \sum_{\mathbf{x} \in X^L} P(\mathbf{x}) \cdot \log_2 P(\mathbf{x}) \quad (6.8)$$

gegeben. X^L entspricht der Menge aller Wortfolgen der Länge L . Die Menge X^0 enthält nur das leere Wort. Für den Vergleich von Sprachäußerungen unterschiedlicher Länge wird die *Entropierate* verwendet. Diese wortweise Entropie lässt sich mit Hilfe der Entropie $H(\mathbf{X})$ aus Gleichung 6.8 durch

$$H_{rate}(\mathbf{X}) = -\frac{1}{L} \sum_{\mathbf{x} \in X^L} P(\mathbf{x}) \cdot \log_2 P(\mathbf{x}) = \frac{H(\mathbf{X})}{L} \quad (6.9)$$

berechnen [JM14]. Analog zur *Entropierate* soll für die Bewertung der generierten Wortfolgen im gewichteten Transduktor T_{AW} die Anzahl der Wörter $L = |\mathbf{y}|$ einer Wortfolge \mathbf{y} zur Normierung verwendet werden. Die normierten Kosten durchgehender Pfade $w[\pi]_{rate}$ sind in diesem Fall durch

$$w[\pi]_{rate} = \frac{w[\pi]}{|\mathbf{y}[\pi]| - 1} \quad (\text{mit } |\mathbf{y}[\pi]| > 1) \quad (6.10)$$

zu bestimmen. Da das Startsymbol $\langle \text{start} \rangle$ immer Bestandteil der Wortfolge \mathbf{y} ist, ihm jedoch durch das Bigramm-Sprachmodell kein Gewicht zugeordnet wird, ist es für die Normierung in den Gleichungen 6.10 und 6.11 nicht berücksichtigt worden. Das

Ergebnis für die N -besten Pfade liefert ein Transduktor mit

$$T_{AW_{N-best}} = \bigoplus_{n=1}^N \arg \operatorname{rank}_n \frac{w[\pi]}{|\mathbf{y}[\pi]| - 1}. \quad (6.11)$$

6.3.3 Microsoft Web N -gram Sprachmodell

Zusätzlich zu dem Bigramm-Sprachmodell erfolgte für die deutsche Lautsprache eine gewichtete Bewertung der generierten Sätze durch den Dienst *Microsoft Web N -gram*. In [WTV⁺10] ist ein Überblick zu dem verwendeten Korpus gegeben. Diese Plattform ermöglicht die Nutzung eines geglätteten Sprachmodells, welches aus realen Daten von vielen verschiedenen Webseiten erstellt wurde. Zur Glättung wird ein *absolute Discounting* verwendet [GNL⁺10], bei dem die abgezogenen Häufigkeitswerte (Discount-Summe) auf die relative Häufigkeit der unbeobachteten N -Gramme verteilt werden. Für die eigenen Anfragen wurde ein N -Gramm-Modell verwendet, welches aus dem sichtbaren Inhalt von Webseiten (HTML-Körper) erstellt wurde. Die Sprachdaten stammten dabei aus dem Jahr 2013. Für dieses stochastische Sprachmodell werden N -Gramme bis zur Ordnung 5 und eine entsprechende Rückfallstrategie verwendet. Zusätzlich können auch kleinere Sprachmodelle genutzt werden, welche aus dem Titel der Webseiten und den Querverweisen aus Hypertextdokumenten generiert wurden. Über den von Microsoft zur Verfügung gestellten webbasierten Dienst erhält man für eine Wortsequenz deren Bewertung. Die Bestimmung des Gesamtgewichts erfolgt hierbei durch $w = \log_{10} P(x_i | x_{i-(N-1)} \dots x_{i-1})$.

Die als Trainingsstichprobe verwendeten Webseiten stammen größtenteils aus dem englischsprachigen Bereich. Das *Microsoft Web N -gram Sprachmodell* kann auch für die deutsche Sprache verwendet werden. Eine explizite Sprachauswahl bei der Dienstanfrage ist nicht möglich. Auch über den Umfang der deutschsprachigen Texte ist nichts bekannt. Insofern kann keine Einschätzung zu der Aussagekraft der Ergebnisse getroffen werden. Bei der Erstellung des Microsoft Sprachmodells wurden die Satzzeichen der geparsten Zeichenfolgen entfernt und Großbuchstaben durch Kleinbuchstaben ersetzt [GNL⁺10]. Dies kann natürlich u. U. problematisch sein. Betrachtet man z. B. die Wörter *verstand* (Verb) und *Verstand* (Substantiv), so stellt man fest, dass diese im unterschiedlichen sprachlichen Kontext verwendet werden. An dieser Stelle ist anzumerken, dass aktuell eine Umstellung des webbasierten Dienstes *Microsoft Web N -gram* erfolgt. Zukünftig können die vergleichbaren *Project Oxford Web Language APIs* [MSW16] verwendet werden.

6.3.4 Ergebnisse der normierten Bewertung von generierten Wortfolgen

Als Ergebnis des Generierungsprozesses zu der in Abb. 6.3 gezeigten Bedeutungsstruktur enthält der Transduktor T_{AW} 57024 mögliche Wortfolgen der deutschen Lautsprache. Diese wurden durch zwei verschiedene Sprachmodelle beurteilt. Anhang C.2 zeigt einen kleinen Ausschnitt der bewerteten Symbolfolgen. Dabei sind in Abhängigkeit vom Bewertungsverfahren jeweils die 25 besten Folgen, die 25 am schlechtesten bewerteten Ausdrücke und 25 Sequenzen aus dem Mittelfeld aufgelistet.

Die Tabelle 6.1 beinhaltet jeweils die Anzahl an grammatisch unstimmgigen Wortfolgen für die N -besten Sprachmodellbewertungen. Dabei wurde manuell entschieden, ob eine Sprachäußerung bezüglich der Zielsprache wohlgeformt ist oder nicht. Das

Sprachmodell	$N = 5$	$N = 10$	$N = 20$	$N = 30$
Bigramm (normiert)	0	2	6	12
Bigramm (gesamt)	2	4	9	15
Microsoft Sprachmodell (normiert)	5	9	18	26
Microsoft Sprachmodell (gesamt)	5	8	18	27

Tabelle 6.1: Anzahl grammatisch unstimmgiger Sätze in der N -besten Liste

Bewertungsergebnis des *Microsoft Web N-gram* Sprachmodells für diese einzelne Stichprobe ist als unbefriedigend zu bezeichnen. Eine allgemeine Eignung für die Bewertung von Wortfolgen der deutschen Sprache kann angezweifelt werden. In der anhand des normierten Gesamtgewichts bestimmten Rangfolge treten gehäuft lange Wortfolgen auf den vorderen Plätzen auf. Diese Sequenzen beinhalten auffällig oft das auch im Englischen verwendeten Sprachzeichen *stand* sowie die bestimmten Artikeln (*der*, *die* und *das*). Die Unigrammwahrscheinlichkeit dieser Sprachzeichen wird vermutlich durch das Microsoft Sprachmodell relativ hoch eingeschätzt. Zusätzlich werden wahrscheinlich nur sehr wenige N-Gramme höherer Ordnung tatsächlich im verwendeten Textkorpus vorhanden sein.

Problematisch bei dem eigenen Bigramm-Sprachmodell ist, dass einige syntaktische Abhängigkeiten durch die niedrige N -Gramm-Ordnung nicht erfasst werden können. Zum Beispiel kann bei der Verwendung von Bigrammen für die Folge „*der wartende Peter*“ kein Zusammenhang von dem Artikel *der* und dem dazugehörigen Subjekt *Peter* berücksichtigt werden. Abhilfe schafft möglicherweise ein besseres statistisches Sprachmodell, welches mit einer geeigneten Lernstichprobe trainiert wurde.

Alternativ kann für eine eingeschränkte Sprachdialoganwendung auch eine geeignete Grammatik zum Einsatz kommen, welche ausschließlich wohlgeformte syntaktische Ausdrücke der Zielsprache berücksichtigt.

Kapitel 7

Zusammenfassung und Ausblick

7.1 Zusammenfassung

Die Untersuchungen zeigen, dass *Äußerungs-Bedeutungs-Transduktoren* erfolgreich zur bidirektionalen Übersetzung zwischen Sprachzeichenfolgen und hierarchischen Bedeutungsrepräsentationen eingesetzt werden können. Folgen von Gebärdensprachzeichen müssen dazu mittels einer beschrifteten Totalordnung repräsentiert werden. Die Klammerstruktur von *MWR-Zeichenketten* ermöglicht die Konstruktion von (gewichteten) *Merkmals-Werte-Relationen*. Bei der Artikulation werden durch den gewählten Linearisierungsprozess alle syntaktisch möglichen Variationen von semantischen Zeichenketten erzeugt. Somit wird erreicht, dass inverse *Äußerungs-Bedeutungs-Transduktoren* die zur Bedeutungsstruktur zugeordneten Sprachäußerungen generieren.

Durch die Verwendung der erstellten semantischen Grammatiken werden auf der Perzeptionsseite drei wesentliche Aufgaben erfüllt:

1. Zuordnung von Bedeutungshypothesen in Abhängigkeit der Interaktionssituation,
2. Generierung einer angemessenen Bedeutungsrepräsentation und
3. die Normalisierung von Datenwerten.

In dieser Arbeit wird eine mögliche Vorgehensweise zur Modellierung von *Äußerungs-Bedeutungs-Transduktoren* vorgestellt. Die dafür verwendeten Grammatiken werden mit Hilfe einer Menge von Äußerungs-Bedeutungs-Paaren erstellt. Eine solche Zuordnung der Lernstichprobe findet man z. B. auch bei dem *Hidden Vector State Modell* [HY05] und dem *Semantic Tuple Classifier* [MGJ⁺09]. Eine Berücksichtigung der vorherrschenden Sprechsituation erfolgt dabei ansatzweise nur beim *Semantic Tuple Classifier*, da hier

zunächst die Art des Dialogakts klassifiziert wird. Diese beiden Verfahren generieren ebenfalls hierarchische Bedeutungsstrukturen und erfassen somit bestehende Relationen zwischen semantischen *Konzepten*.

Im Allgemeinen wird für das Training stochastischer Modelle eine große, domänenspezifische Lernstichprobe benötigt. Einige Verfahren verwenden Wortklassen, die jeweils eine Vielzahl möglicher Werte zusammenfassen. Auch zur Modellierung von *Äußerungs-Bedeutungs-Transduktoren* wird das Prinzip der Generalisierung angewandt. Die dabei eingesetzten Platzhalter sind jedoch universeller. Sie können neben Wortklassen auch verschiedene Ausprägungen *semantischer Anker* oder sogar ganze *semantische Schemata* determinieren. Auf diese Weise wird der Umfang der benötigten Sprachdaten reduziert. Zudem stellt die Verwendung von Platzhaltern auch die Grundlage dafür dar, dass der Aufwand zur Erstellung von Grammatiken beherrschbar ist. Bereits anhand der ermittelten Sprachäußerungen zu den hier erstellten *Elementargrammatiken* ist ersichtlich, dass eine große Anzahl unterschiedlicher Sprachzeichenfolgen zum Ausdruck einer konkreten Absicht möglich sind. Trotz der Möglichkeit zur Automatisierung von Abläufen bedarf es manueller Eingriffe während des Modellierungsprozesses. Ursache dafür ist das erforderliche linguistische Wissen, welches für die Zuordnung von semantischen Ankern und Platzhaltern zu einzelnen Sprachzeichen oder Sprachzeichenfolgen benötigt wird. Für die Berücksichtigung des grammatischen Geschlechts sind zusätzliche sprachorientierte Platzhalter für Pronomen und Artikel erforderlich, welche jedoch regelbasiert in eine Grammatik integriert werden können. Platzhaltertransduktoren sind hier sehr allgemein modelliert worden. Dafür werden verschiedene Lexika (z. B. für Flexionen und Synonyme) und domänenspezifische Wissensmodelle benötigt. Dieser Ansatz hat jedoch wesentliche Vorteile. Neben der universellen Verwendbarkeit dieser Automaten und einem geringeren manuellen Modellierungsaufwand wird eine erhöhte Robustheit des Interpretationsprozesses erreicht.

Werden inverse *Äußerungs-Bedeutungs-Transduktoren* hingegen für den Artikulationsvorgang eingesetzt, so generieren diese auch grammatisch unstimmmige Sprachzeichenfolgen. Durch ein geeignetes stochastisches Sprachmodell der Zielsprache ist eine Bestimmung der Rangfolge artikulierter Wortfolgen möglich. Ein Nachweis der Eignung dieser Sprachmodellbewertung kann nicht erbracht werden. Dafür sind u. a. ein größerer Korpus, eine größere Anzahl semantischer Grammatiken sowie N-Gramm-Modelle höherer Ordnung notwendig. Der zur Modellerstellung verwendete Textkorpus sollte möglichst zu dem beschränkten Anwendungsgebiet passen. Frei verfügbare monolinguale Daten, wie z. B. *Europarl* (European Parliament Proceedings Parallel Corpus 1996-2011) [Koe05, Sta16] oder *News Crawl* (Nachrichtenartikel) [Sta16] sind in der Regel sehr spezifisch bezüglich der verwendeten Sprache. Aus diesem Grund wird in den meisten Fällen noch zusätzlicher Aufwand benötigt, um geeignetes Textmaterial zu gewinnen.

7.2 Weitere Untersuchungs- und Anwendungsmöglichkeiten

Durch diese Arbeit ergibt sich direkt eine Reihe weiterer, interessanter Forschungs- und Entwicklungsmöglichkeiten.

Mögliche Vorteile der Anwendung von *Äußerungs-Bedeutungs-Transduktoren* und *Merkmal-Werte-Relationen* in intelligenten dynamischen Signalverarbeitungssystemen können erst durch vergleichende praktische Untersuchungen konkret bestimmt werden. Hierfür sind u. a. auch die Einbeziehung von Analyse, Synthese und Verhaltenssteuerung sowie einer bidirektionalen Informationsverarbeitung notwendig.

In diesem Zusammenhang ist z. B. auch eine Gebärdenspracherkennung und -synthese nötig. Aktuelle Untersuchungen im Bereich der sprecherunabhängigen, statistischen Gebärdenspracherkennung [DSD⁺08, CHB11, KFN15] sind vielversprechend. Problematisch ist hier aber vor allem die Verfügbarkeit notwendiger annotierter Sprachdaten zum Modelltraining. Die in dieser Arbeit verwendeten Gebärdensprachäußerungen beinhalten nur einfache Gebärden. Diffizil könnte beispielsweise das Auftreten klassifizierender Morpheme in komplexen Gebärden sein. Weiterhin müssen grammatisch bedingte Modifikationen einer Gebärdenform als einzelne Lexikoneinheiten betrachtet werden, was zu einem starken Anstieg der benötigten Sprachdatensammlung führt. Für eine Perzeption von Gebärdensprachäußerungen sind noch viele weitere Untersuchungen nötig. Dabei ist auch eine Zusammenarbeit mit Gebärdensprachlinguisten erforderlich.

Akustische Informationen müssen für gehörlose Menschen visualisiert werden. Eine Sprachsynthese erfolgt durch Verwendung von Gebärdensprachavataren (siehe auch [KNH11]). Dazu muss eine artikulierte Glossenfolge zunächst in eine animationsorientierte Notation von Gebärdensprachzeichen überführt werden, welche für die Ansteuerung von Avataren erforderlich ist.

Klassische multilinguale Systemmodelle für Lautsprachen, wie das VOYAGER-System [GFG⁺95] und das KIT-Sprachdialogsystem [XAN03], nutzen zur semantischen Interpretation sprachenabhängige Grammatiken. Aktuelle Forschungsvorhaben untersuchen Möglichkeiten der Portierung sprachspezifischer Modelle mit Hilfe maschineller Übersetzungsverfahren. Dabei können zwei verschiedene Ansätze verfolgt werden. Zum einen werden die Sprachdaten für das Training der semantischen Modelle übersetzt. Somit erfolgt die Interpretation in Abhängigkeit der erkannten natürlichen Sprache des Benutzers. Alternativ wird bei der Perzeption eine maschinelle Übersetzung der erkannten Sprachzeichenfolge in die Zielsprache durchgeführt, für welche bereits trainierte Modelle vorhanden sind [JBL13, CHG⁺16].

Eine direkte Übersetzung der erkannten Gebärdensprachäußerung in eine Wortfolge der Lautsprache wäre theoretisch auch hier möglich. Gleichzeitig kann auch auf der Syntheseseite eine Übersetzung der artikulierten Wortfolge in eine animationsorientierte

Gebärdensprachnotation erfolgen. Informationen zur statistischen maschinellen Übersetzung von Gebärdensprachezeichen findet man beispielsweise in [Mor08, KM09, SSN12]. Auf dem gesamten Gebiet der Gebärdensprachverarbeitung besteht genügend Potenzial für eine Vielzahl von Forschungsvorhaben zur Entwicklung neuer Modelle und Konzepte oder zur Optimierung bestehender Ansätze. Für eine Berücksichtigung von Gebärdensprachen in Sprachdialogsystemen ist aus heutiger Sicht aber vor allem ein enormer Entwicklungsaufwand nötig.

Für weitere Untersuchungen ist es sinnvoll, einen Bezug zu einer konkreten, technischen Anwendung herzustellen. Anhand eines zu erstellenden Systemmodells können mögliche ausführbare Systemaktionen, Datenbankstrukturen und anwendungsabhängiges Wissen genau spezifiziert werden. Dadurch ist eine Bestimmung *semantischer Schemata* und *semantischer Anker* möglich. Weiterhin kann anhand der festgelegten Funktionalität des Systems eine genaue Identifikation von Benutzerzielen erfolgen. Mit diesen Informationen ist man in der Lage, ein funktionsfähiges System zu simulieren sowie *Wizard-of-Oz Experimente* zu planen und durchzuführen. Auf diesem Weg ist eine anwendungsorientierte Erstellung mehrerer *mikrolokaler Grammatiken* realisierbar. Hierzu ist eine weitere Automatisierung des Modellierungsprozesses zweckmäßig. Dafür können beispielsweise Skripte zu Extraktion von Zeichen aus Wissensquellen und zur Grammatikkonstruktion geschrieben werden. Sinnvoll ist zudem die Erstellung einer Entwicklungsumgebung zur Modellierung von Grammatiken. Diese stellt z. B. grafische Werkzeuge zur Manipulation von Automaten zur Verfügung, ermöglicht eine Verwaltung erstellter Objekte, die Anzeige von Automatengraphen und *Merkmal-Werte-Relationen* sowie eine Ausführung von Automatenoperationen. Eine solche Software ist besonders hilfreich, wenn mehrere Entwickler oder aber auch Gehörlose und Gebärdensprachlinguisten bei dem Modellierungsprozess beteiligt sind.

Weiterhin müssen Strategien entwickelt werden, wie *semantische Schemata* modularisiert werden können. So wäre auch der Einsatz von optionalen Platzhaltern in Grammatiken vorstellbar. Außerdem treten ggf. komplexe Sprachäußerungen auf, in denen gleich mehrere Teilaufgaben vermittelt werden. Unter Umständen ist dabei auch die Reihenfolge der Abarbeitung dieser Teilziele bedeutungsunterscheidend. In diesem Fall müssen *Merkmal-Werte-Relationen* erstellt werden, welche auch Abläufe repräsentieren.

Ist eine Anwendung prototypisch entwickelt, so ist man in der Lage, auch Feld- bzw. Pilottest mit Probanden durchzuführen. Aus diesen Erkenntnissen kann ebenfalls noch eine Adaption der semantischen Grammatiken resultieren.

Komplexere technische Systeme müssen zeitveränderliches domänenspezifisches Wissen verarbeiten können. So ist zu untersuchen, wie eine Adaption vorhandener Modelle im laufenden Betrieb realisiert werden kann. Dabei müssen eventuell Datenwerte in Platzhaltern aktualisiert werden, wenn neue Datenbankeinträge zur Verfügung stehen. Ein weiter wichtiger Punkt, welcher in dieser Arbeit nicht betrachtet wurde, besteht

in der Möglichkeit des Lernens neuer Äußerungs-Bedeutungs-Paare. So kann es z. B. immer wieder vorkommen, dass unbekannte Sprachzeichenfolgen beobachtet werden. Ein erster Ansatz zum autonomen Lernen wurde in [KWL⁺14] präsentiert, in der das Eintreten einer ereignishaften Situation zu einer Anpassung des semantischen Modells führt.

In dieser Arbeit wurden nur *Äußerungs-Bedeutungs-Transduktoren* zu einem semantischen Schema erzeugt. Aus diesem Grund gibt es keine praktischen Erfahrungen bezüglich eines gleichzeitigen Einsatzes bzw. der Vereinigung mehrerer Transduktoren für Interpretation und Artikulation. Weiterhin sind Untersuchungen bezüglich der Rückweisung unsicherer Hypothesen des Interpretationsergebnisses nötig. Dabei sind neben stochastischen semantischen Grammatiken auch gewichtete Worthypotheseographen von einem Spracherkenner zu verwenden.

Künftige Forschungen müssen auch die Verhaltenssteuerung einbeziehen. Dies ermöglicht u. a. die Berücksichtigung der Interaktionssituation bzw. der Erwartung beim Interpretationsprozess. Zudem müssen beispielsweise Unifikationsalgorithmen zur Vereinigung erkannter stochastischer *Merkmal-Werte-Relationen* und Möglichkeiten der Ersetzung von *referenziellen semantischen Anker*n untersucht werden. In diesem Zusammenhang ist auch die automatische Erweiterung einer MWR um die ID-Schicht nötig, wodurch eine Identifikation von Datenbankobjekten erreicht wird.

Wie bereits im Abschnitt 7.1 angemerkt, bedarf es noch weiterer praktischer Tests bezüglich des Einsatzes von stochastischen Sprachmodellen bei der Artikulation. Aufschlussreich können auch Untersuchungen bezüglich der Verständlichkeit und Akzeptanz von grammatisch unstimmgigen, synthetisierten Laut- und Gebärdensprachäußerungen sein.

Viele der hier aufgeführten Aspekte setzen einen vorhandenen Prototypen mit entsprechender Grundfunktionalität voraus. Weiterführende Betrachtungen sollten deshalb anhand eines konkreten technischen Experimentiersystems erfolgen. Auch aufgrund der fortschreitenden Miniaturisierung technischer Systeme können Dialogsysteme in unterschiedlichsten Bereichen eingesetzt werden. Mögliche Anwendungsgebiete sind beispielsweise:

- im *Smart-Home*-Bereich
 - Hausautomationstechnik (Licht, Heizung, Sicherheit, ...)
 - Assistenz- und Informationssysteme (altersgerechte / persönliche Assistenzsysteme, Kommunikationshilfen, „intelligenter Badspiegel“, Roboter)
 - Steuerung von Multimedia-Geräten (Film, Musik und Spiele)
 - intelligente Haushaltsgeräte

- im mobilen Einsatz
 - Sprachbedienung im Auto
 - Informationssysteme für Touristen
 - Assistenz- und Informationssysteme für Smartphone
- im Bereich öffentlicher und kommerzieller Dienste
 - Informations- und Auskunftssysteme (Behörden, Verkehr, ...)
 - automatisierte Telefondienste (Anrufklassifikation, Aufnahme von Kundendaten, ...)
 - inner- und zwischenbetriebliche Anwendungen

Eine Interpretation von Benutzerverhalten erfolgte in dieser Arbeit nur anhand der Sprachzeichenfolge. In einem multimodalen System ist zudem auch beobachtbares nicht-sprachliches Verhalten modalitätsspezifisch zu perzipieren. Simultan zum sprachlichen Verhalten ermittelte semantische Informationen können ergänzend, bestätigend oder kontradiktorisch sein. Eine Verhaltenssteuerung intelligenter Systeme muss deshalb auch in der Lage sein, monomodale Interpretationshypothesen zu unifzieren.

Letztendlich sollten auch die Möglichkeiten des Einsatzes von *Petrimetztransduktoren* [LH13, LHW14] oder anderer erweiterter Automatenmodelle zur semantischen und pragmatischen Verarbeitung theoretisch und gegebenenfalls auch praktisch weiter untersucht werden.

Anhang A

Verwendetes Sprachmaterial

A.1 Lautsprachäußerungen zur Modellierung der Elementargrammatik

<there> = dort drüben, da drüben, dort, da

1. <there> steht Peter und wartet.
2. Peter steht <there> und wartet.
3. Peter steht wartend <there>.
4. Wartend steht Peter <there>.
5. Der wartende Peter steht <there>.
6. <there> steht der wartende Peter.
7. <there> wartet der stehende Peter.
8. Der stehende Peter wartet <there>.
9. Stehend wartet Peter <there>.
10. Peter wartet stehend <there>.
11. Peter steht und wartet <there>.
12. Peter wartet und steht <there>.
13. <there> wartet und steht Peter.
14. <there> steht und wartet Peter.
15. Der Peter steht und wartet <there>.
16. Der Peter wartet und steht <there>.
17. <there> wartet und steht der Peter.

18. <there> steht und wartet der Peter.
19. <there> steht der Peter und wartet.
20. Der Peter steht <there> und wartet.
21. Der Peter steht wartend <there>.
22. Wartend steht der Peter <there>.
23. Stehend wartet der Peter <there>.
24. Der Peter wartet stehend <there>.

A.2 Zusätzliche lautsprachliche Äußerungen

Nachfolgend sind weitere Sprachäußerungen zu der erweiterten Elementargrammatik aufgeführt. Diese sind allerdings nicht zur Modellierung verwendet worden. Im Gegensatz zum Anhang A.1 beinhalten die Wortfolgen Wiederholungen von semantischen Konzepten und teilweise zusätzlich das Personalpronomen *er*.

1. Peter steht <there>. Er wartet.
2. Peter steht <there>. Er wartet <there>.
3. Peter steht <there>. Peter wartet.
4. Peter steht <there>. Peter wartet <there>.
5. Peter steht <there>. <there> wartet Peter.
6. Peter steht <there>. <there> wartet er.
7. Peter steht. Peter wartet <there>.
8. Peter steht. Er wartet <there>.
9. Peter steht. <there> wartet Peter.
10. Peter steht. <there> wartet er.
11. <there> steht Peter. Peter wartet <there>.
12. <there> steht Peter. Er wartet <there>.
13. <there> steht Peter. <there> wartet Peter.
14. <there> steht Peter. <there> wartet er.
15. Peter wartet <there>. Peter steht <there>.
16. Peter wartet <there>. Er steht <there>.
17. Peter wartet <there>. Peter steht.
18. Peter wartet <there>. Er steht.

19. Peter wartet <there>. <there> steht Peter.
20. Peter wartet <there>. <there> steht er.
21. Peter wartet. Er steht <there>.
22. Peter wartet schon. Er steht <there>.
23. Peter wartet. Peter steht <there>.
24. Peter wartet schon. Peter steht <there>.
25. Peter wartet. <there> steht er.
26. Peter wartet. <there> steht Peter.
27. Peter wartet schon. <there> steht er.
28. Peter wartet schon. <there> steht Peter.
29. <there> wartet Peter. Peter steht <there>.
30. <there> wartet Peter. Er steht <there>.
31. <there> wartet Peter. <there> steht Peter.
32. <there> wartet Peter. <there> steht er.
33. Peter steht <there> und er wartet.
34. Peter steht <there> und er wartet <there>.
35. Peter steht <there> und Peter wartet.
36. Peter steht <there> und Peter wartet <there>.
37. Peter steht <there> und <there> wartet Peter.
38. Peter steht <there> und <there> wartet er.
39. Peter steht und Peter wartet <there>.
40. Peter steht und er wartet <there>.
41. Peter steht und <there> wartet Peter.
42. Peter steht und <there> wartet er.
43. <there> steht Peter und Peter wartet <there>.
44. <there> steht Peter und er wartet <there>.
45. <there> steht Peter und <there> wartet Peter.
46. <there> steht Peter und <there> wartet er.
47. Peter wartet <there> und Peter steht <there>.
48. Peter wartet <there> und er steht <there>.
49. Peter wartet <there> und Peter steht.
50. Peter wartet <there> und er steht.

51. Peter wartet <there> und <there> steht Peter.
52. Peter wartet <there> und <there> steht er.
53. Peter wartet und er steht <there>.
54. Peter wartet schon und er steht <there>.
55. Peter wartet und Peter steht <there>.
56. Peter wartet schon und Peter steht <there>.
57. Peter wartet und <there> steht er.
58. Peter wartet und <there> steht Peter.
59. Peter wartet schon und <there> steht er.
60. Peter wartet schon und <there> steht Peter.
61. <there> wartet Peter und Peter steht <there>.
62. <there> wartet Peter und er steht <there>.
63. <there> wartet Peter und <there> steht Peter.
64. <there> wartet Peter und <there> steht er.

<there > = dort drüben, da drüben, dort, da

A.3 **Verwendete Glossennotationen zur Modellierung der Gebärdensprachgrammatiken**

1. PETER STEHEN WARTEN DORT
2. PETER WARTEN STEHEN DORT
3. PETER DORT STEHEN WARTEN
4. DORT PETER WARTEN STEHEN
5. DORT PETER STEHEN WARTEN
6. DORT PETER STEHEN SELBST WARTEN
7. DORT STEHEN WARTEN WER PETER
8. DORT STEHEN WARTEN SELBST PETER
9. DORT WARTEN STEHEN SELBST PETER
10. PETER STEHEN WARTEN WO DORT
11. PETER STEHEN DORT STEHEN *

* Die Sprachäußerung 11 wird nicht für die mikrolokale Grammatik verwendet.

A.4 Gebärdensprachtranskriptionen

Die Gebärdensprachäußerungen wurden von PIA STEINBACH an der *Westsächsischen Hochschule Zwickau* getätigt, als Video aufgezeichnet und anschließend von ANITA JUNGHANNS transkribiert.

verwendete Abkürzungen/Symbole:

Ao → Augenbrauen oben

Au → Augenbrauen unten

[Zz] → Zähne zeigen

[Zr] → Zunge raus

... → Pause

1. Mimik: $\left(\begin{array}{l} \text{neutral} \\ \text{PETER STEHEN WARTEN} \\ \text{peter} \qquad \qquad \text{warten} \end{array} \right.$

2. Mimik: $\left(\begin{array}{l} \text{neutral} \\ \text{PETER WARTEN STEHEN} \\ \text{peter} \quad \text{warten} \end{array} \right.$

3. Mimik: $\left(\begin{array}{l} \text{neutral} \qquad \qquad \qquad \text{Ao} \\ \text{PETER STEHEN WARTEN DORT} \\ \text{peter} \quad \text{stehen} \qquad \qquad \text{[Zz]} \end{array} \right.$

4. Mimik: $\left(\begin{array}{l} \text{neutral} \qquad \qquad \qquad \text{Ao} \\ \text{PETER STEHEN WARTEN DORT} \\ \text{peter} \quad \text{stehen} \quad \text{warten} \quad \text{[Zz]} \end{array} \right.$

5. Mimik: $\left(\begin{array}{l} \text{neutral} \qquad \qquad \qquad \text{Ao} \\ \text{PETER WARTEN STEHEN ... DORT} \\ \text{peter} \quad \text{warten} \qquad \qquad \text{[Zz]} \end{array} \right.$

6. Mimik: $\left(\begin{array}{c} \text{neutral} \quad \text{Ao} \quad \text{neutral} \\ \text{PETER STEHEN... DORT... STEHEN} \\ \text{peter} \quad \quad \quad [\text{Zz}] \end{array} \right.$
7. Mimik: $\left(\begin{array}{c} \text{neutral} \quad \text{Ao} \quad \text{neutral} \\ \text{PETER ... DORT STEHEN WARTEN} \\ \text{peter} \quad \quad [\text{Zz}] \quad \quad \quad \text{warten} \end{array} \right.$
8. Mimik: $\left(\begin{array}{c} \text{neutral} \quad \text{Ao} \quad \text{neutral} \\ \text{PETER DORT STEHEN... WARTEN} \\ \text{peter} \quad [\text{Zz}] \quad \quad \quad \text{warten} \end{array} \right.$
9. Mimik: $\left(\begin{array}{c} \text{neutral} \\ \text{PETER ... STEHEN... WARTEN DORT} \\ \text{peter} \quad \quad \quad \text{warten} \quad [\text{Zz}] \end{array} \right.$
10. Mimik: $\left(\begin{array}{c} \text{Ao} \quad \text{neutral} \\ \text{DORT PETER WARTEN STEHEN} \\ [\text{Zz}] \quad \text{peter} \quad \text{warten} \quad [\text{Zr}] \end{array} \right.$
11. Mimik: $\left(\begin{array}{c} \text{Ao neutral} \\ \text{DORT PETER... STEHEN WARTEN} \\ [\text{Zz}] \quad \text{peter} \quad \quad \quad \text{warten} \end{array} \right.$
12. Mimik: $\left(\begin{array}{c} \text{neutral} \quad \text{Ao} \quad \text{neutral} \\ \text{DORT... PETER STEHEN WARTEN} \\ [\text{Zz}] \quad \quad \text{peter} \quad \quad \quad \text{warten} \end{array} \right.$
13. Mimik: $\left(\begin{array}{c} \text{Au} \quad \quad \quad \text{Ao} \quad \text{neutral} \\ \text{DORT PETER STEHEN SELBST WARTEN} \\ [\text{Zz}] \quad \text{peter} \quad \quad \quad \text{selbst} \quad \text{warten} \end{array} \right.$

14. Mimik: $\left(\begin{array}{c} \text{neutral} \quad \text{Ao} \quad \text{neutral} \\ \text{PETER STEHEN SELBST WARTEN} \\ \text{peter} \quad \quad \text{selbst} \quad \text{warten} \end{array} \right.$
15. Mimik: $\left(\begin{array}{c} \text{Au} \quad \quad \quad \text{Ao} \quad \text{neutral} \\ \text{DORT STEHEN WARTEN WER ... PETER} \\ [\text{Zz}] \quad \text{stehen} \quad [\text{Zr}] \quad \text{wer} \quad \text{peter} \end{array} \right.$
16. Mimik: $\left(\begin{array}{c} \text{Au} \quad \quad \quad \text{Ao} \quad \text{neutral} \\ \text{DORT STEHEN WARTEN ... SELBST PETER} \\ [\text{Zz}] \quad \quad \quad \text{warten} \quad \text{selbst} \quad \text{peter} \end{array} \right.$
17. Mimik: $\left(\begin{array}{c} \text{Au} \quad \quad \quad \text{Ao} \quad \text{neutral} \\ \text{STEHEN WARTEN SELBST PETER} \\ \quad \quad \quad \text{warten} \quad \text{selbst} \quad \text{peter} \end{array} \right.$
18. Mimik: $\left(\begin{array}{c} \text{Au} \quad \quad \quad \text{Ao} \quad \text{neutral} \\ \text{DORT WARTEN STEHEN ... SELBST PETER} \\ \quad \quad \quad \text{warten} \quad [\text{Zr}] \quad \text{selbst} \quad \text{peter} \end{array} \right.$
19. Mimik: $\left(\begin{array}{c} \text{Au} \quad \quad \quad \text{Ao} \quad \text{neutral} \\ \text{STEHEN WARTEN WER ... PETER} \\ \quad \quad \quad \text{warten} \quad \text{wer} \quad \text{peter} \end{array} \right.$
20. Mimik: $\left(\begin{array}{c} \text{neutral} \quad \quad \quad \text{Ao} \\ \text{PETER STEHEN WARTEN DORT} \\ \text{peter} \quad \quad \quad [\text{Zz}] \end{array} \right.$
21. Mimik: $\left(\begin{array}{c} \text{Au} \quad \quad \quad \text{Ao} \quad \text{neutral} \\ \text{PETER ... STEHEN WARTEN WO DORT} \\ \text{peter} \quad \quad [\text{Zr}] \quad \quad \quad \text{wo} \quad [\text{Zz}] \end{array} \right.$

Anhang B

Automatengraphen erstellter Transduktoren

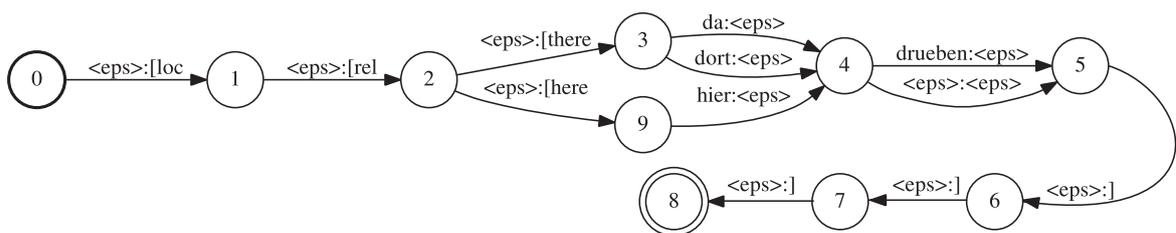


Abb. B.1: FST für Platzhalter „loc_rel“ (Lautsprache)

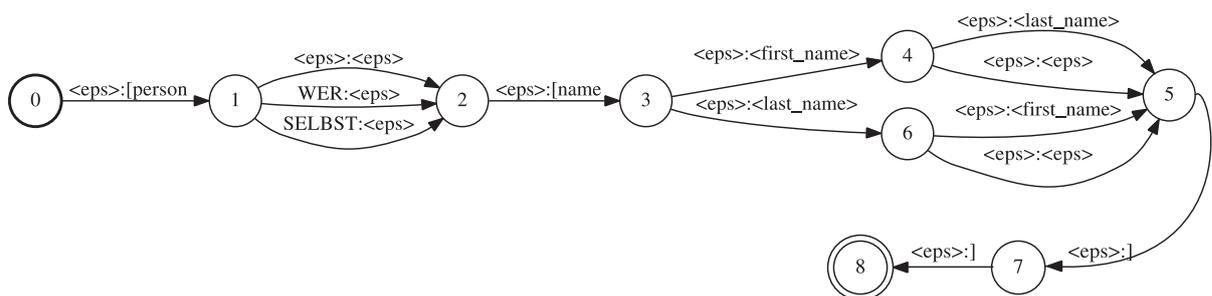


Abb. B.2: FST für Platzhalter „name“ (DGS)

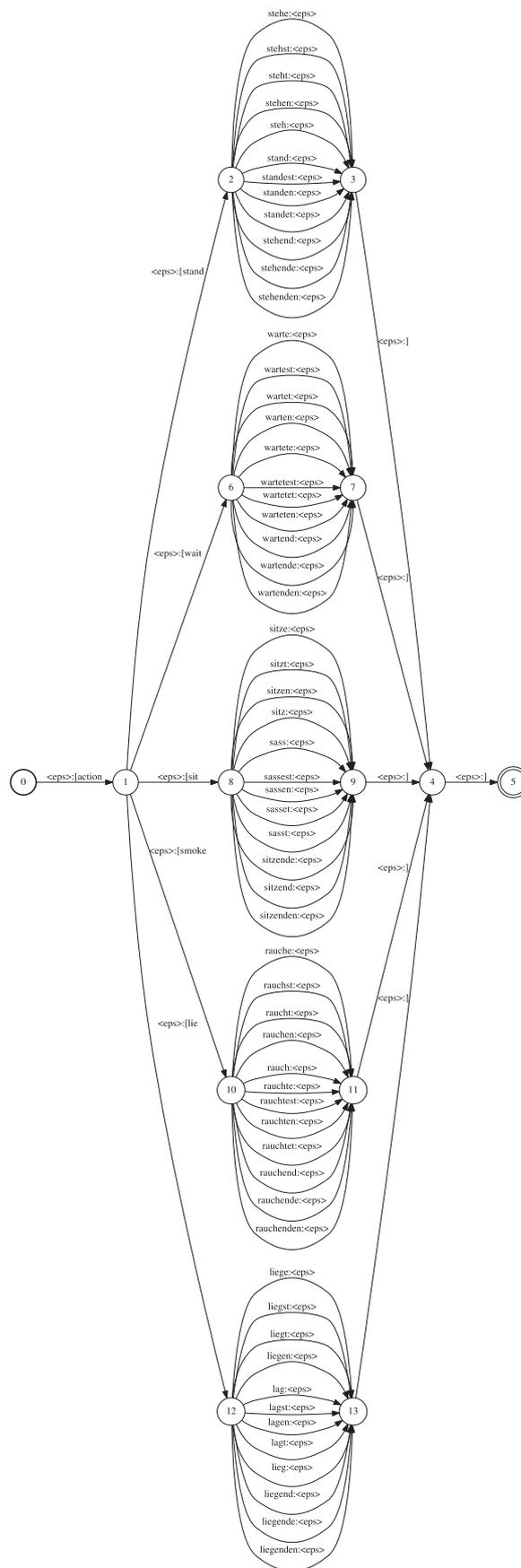


Abb. B.3: FST für Platzhalter „action“ (Lautsprache)

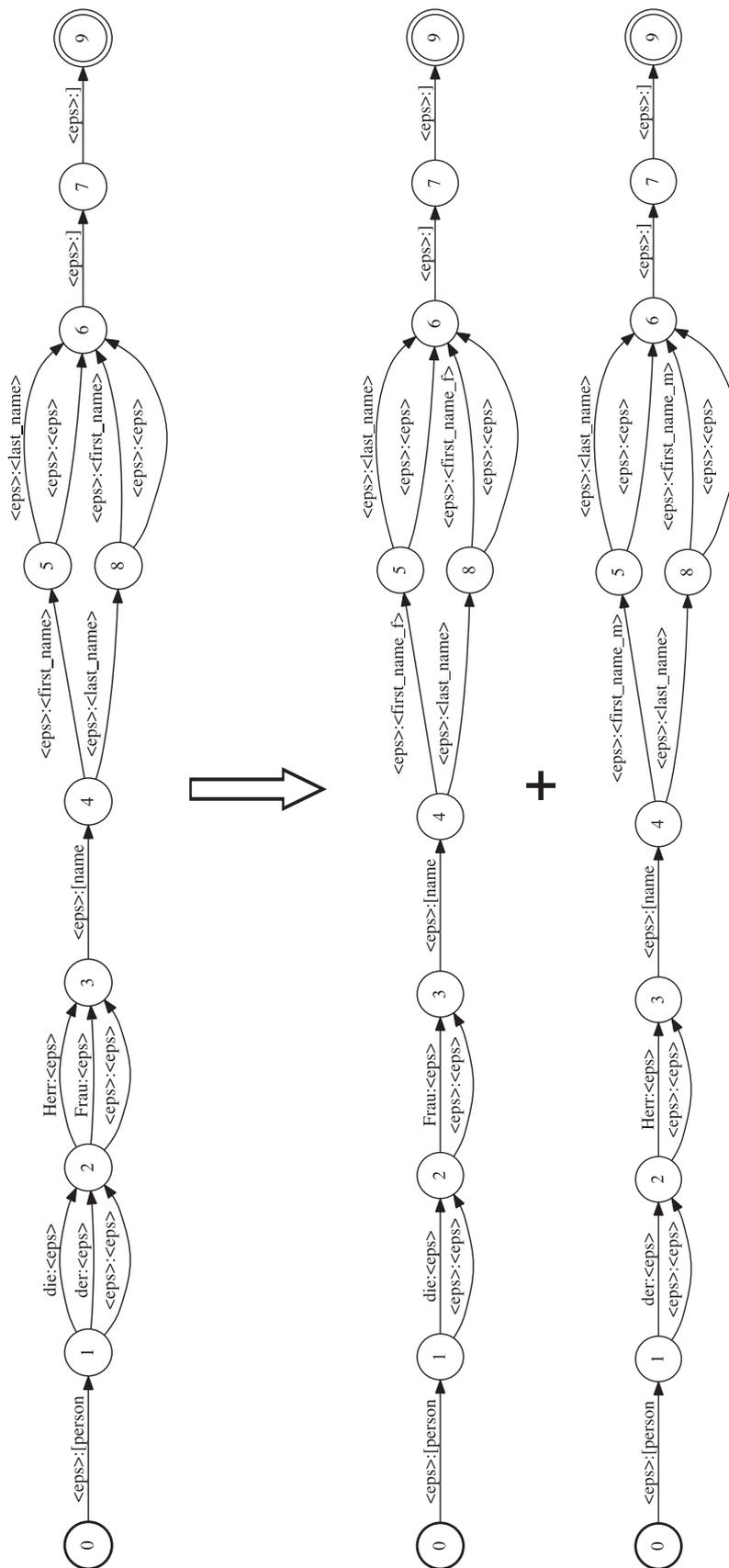


Abb. B.4: Unterteilung des Platzhalters "name" (Lautsprache) unter Berücksichtigung des Geschlechts einer Person

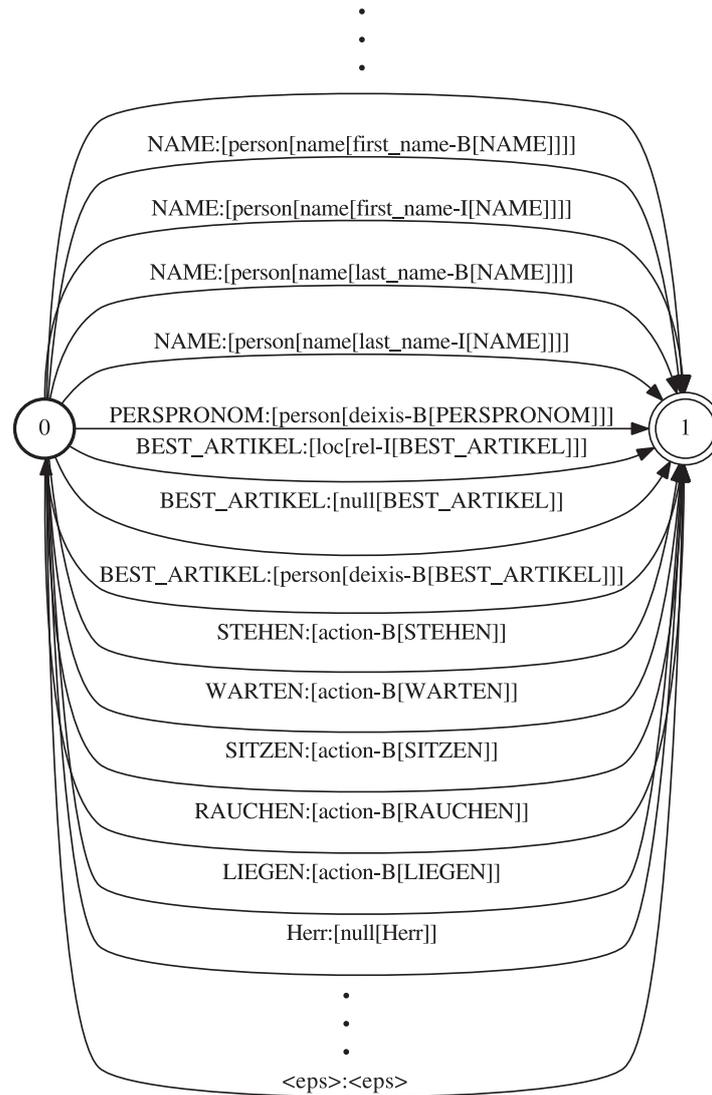


Abb. B.6: Ausschnitt des Transduktors T_{G2K}

Anhang C

Generierte Sprachzeichenfolgen

C.1 Artikulierte Glossensequenzen

Die semantische Interpretation zu der Wortfolge *“Wartend steht Peter dort drüben.”* liefert die in Abbildung 6.3 visualisierte Bedeutungsrepräsentation. Diese wurde mit dem *inversen Äußerungs-Bedeutungs-Transduktor* für die Deutsche Gebärdensprache artikuliert. Als Ergebnis erhält man die folgenden ungewichteten Glossenfolgen:

1. WO DORT WER PETER WARTEN SELBST STEHEN
2. WO DORT WER PETER WARTEN STEHEN
3. WO DORT SELBST PETER WARTEN SELBST STEHEN
4. WO DORT SELBST PETER WARTEN STEHEN
5. WO DORT PETER WARTEN SELBST STEHEN
6. WO DORT PETER WARTEN STEHEN
7. WO DORT WARTEN STEHEN WER PETER
8. WO DORT WARTEN STEHEN SELBST PETER
9. WO DORT WARTEN STEHEN PETER
10. WER PETER WO DORT WARTEN STEHEN
11. WER PETER DORT WARTEN STEHEN
12. WER PETER WARTEN STEHEN WO DORT
13. WER PETER WARTEN STEHEN DORT
14. SELBST PETER WO DORT WARTEN STEHEN
15. SELBST PETER DORT WARTEN STEHEN
16. SELBST PETER WARTEN STEHEN WO DORT
17. SELBST PETER WARTEN STEHEN DORT

18. DORT WER PETER WARTEN SELBST STEHEN
19. DORT WER PETER WARTEN STEHEN
20. DORT SELBST PETER WARTEN SELBST STEHEN
21. DORT SELBST PETER WARTEN STEHEN
22. DORT PETER WARTEN SELBST STEHEN
23. DORT PETER WARTEN STEHEN
24. DORT WARTEN STEHEN WER PETER
25. DORT WARTEN STEHEN SELBST PETER
26. DORT WARTEN STEHEN PETER
27. PETER WO DORT WARTEN STEHEN
28. PETER DORT WARTEN STEHEN
29. PETER WARTEN STEHEN WO DORT
30. PETER WARTEN STEHEN DORT

C.2 Ergebnisse der Sprachmodellbewertung

Anhand der in Grafik 6.3 dargestellten Bedeutungsrepräsentation zu dem semantisch interpretierten Satz *“Wartend steht Peter dort drüben.”* wurden die Ergebnisse des Artikulationsprozesses durch Sprachmodelle bewertet. Beispielhaft sind nachfolgend aus den 57024 möglichen Wortfolgen jeweils ein paar Sequenzen aufgelistet.

C.2.1 Rangfolge durch normierte Bewertung (Bigramm-Sprachmodell)

1. Peter wartet dort drueben und steht
2. Peter wartet und steht dort drueben
3. Peter wartet da drueben und steht
4. der Peter wartet dort drueben und steht
5. Peter wartet und steht da drueben
6. der Peter wartet und steht dort drueben
7. der Peter wartet da drueben und steht
8. der Peter wartet und steht da drueben
9. Peter wartet dort drueben und steh
10. Peter wartet dort drueben und stehen
11. der wartende Peter steht dort drueben

12. Peter wartet und steht dort
13. Peter wartet stehend da drueben
14. Peter wartet und steht da
15. Peter wartet stehend dort drueben
16. der wartende Peter steht da drueben
17. der Peter wartet dort drueben und steh
18. der Peter wartet dort drueben und stehen
19. Peter wartet da drueben und steh
20. Peter wartet da drueben und stehen
21. der Peter wartet und steht dort
22. der Peter wartet stehend da drueben
23. der Peter wartet und steht da
24. der Peter wartet stehend dort drueben
25. Peter wartet steht dort drueben

⋮

25800. die warte die Frau Peter standen da drueben
25801. wartest stehen der Peter dort
25802. der wartend der Frau Peter stehend da drueben
25803. der warten der Frau Peter stehend da drueben
25804. da drueben wartetet der stehend die Frau Peter
25805. da drueben wartend der standen die Frau Peter
25806. da drueben wartend der stand die Frau Peter
25807. da drueben warten der standen die Frau Peter
25808. da drueben warten der stand die Frau Peter
25809. dort wartende die stehende Peter
25810. Peter warten da und stehenden
25811. Peter warten da und stehende
25812. Peter warten da und stehend
25813. Peter warten da und standet
25814. Peter warten da und standen
25815. Peter warten da und standest
25816. Peter warten da und stand
25817. Peter warten da und stehst
25818. Peter warten da und stehe

25819. wartend stehende der Frau Peter da drueben
25820. dort drueben wartend der steht die Peter
25821. dort drueben warten der steht die Peter
25822. die wartetest die Frau Peter stehen da drueben
25823. Frau Peter wartenden stehst da drueben
25824. da drueben wartest das steht die Peter
:

57000. wartet standen die Peter dort
57001. wartet stand die Peter dort
57002. wartet stehende die Peter dort
57003. wartet standet der Frau Peter da
57004. wartet standest der Frau Peter da
57005. wartet stehenden der Frau Peter da
57006. wartet steh der Frau Peter da
57007. wartet stehst der Frau Peter da
57008. wartet stehe der Frau Peter da
57009. wartet standen der Frau Peter da
57010. wartet stand der Frau Peter da
57011. wartet steht Frau Peter da
57012. wartet stehende der Frau Peter da
57013. wartet stehen der Frau Peter da
57014. wartet standet der Frau Peter dort
57015. wartet standest der Frau Peter dort
57016. wartet stehenden der Frau Peter dort
57017. wartet steh der Frau Peter dort
57018. wartet stehst der Frau Peter dort
57019. wartet stehe der Frau Peter dort
57020. wartet standen der Frau Peter dort
57021. wartet stand der Frau Peter dort
57022. wartet stehende der Frau Peter dort
57023. wartet stehen der Frau Peter dort
57024. wartet steht Frau Peter dort

C.2.2 Rangfolge der Gesamtbewertung durch das Microsoft Web N-gram Sprachmodell

1. Peter warte und steht dort
2. Peter warte und steht da
3. Peter warte stand da
4. Peter warten und steht dort
5. Peter warte und stehen da
6. dort wartet und steht Peter
7. da wartet und steht Peter
8. Peter warten und steht da
9. dort warten und steht Peter
10. Peter warte steht da
11. dort wartet das steht Peter
12. Peter warte und stehe da
13. da wartet das steht Peter
14. da warten und steht Peter
15. dort wartete und steht Peter
16. Peter warte dort und steht
17. Peter wartet und steht dort
18. Peter warte stehen da
19. dort wartet der stand Peter
20. dort warteten und steht Peter
21. Peter warte und stehen dort
22. Peter warten und stehen da
23. Peter warte dort und stand
24. da wartet der stand Peter
25. dort wartet der steht Peter

⋮

25800. die wartest der Herr Peter steh da
25801. dort wartest das stand die Peter
25802. da wartetet und standen der Herr Peter
25803. das wartetet Frau Peter standen dort
25804. dort wartende Herr Peter und stehe

25805. da wartende der Herr Peter und standen
25806. warte stehenden der Peter dort drueben
25807. warte standen der Herr Peter da drueben
25808. die Peter wartende da drueben und stehe
25809. da drueben wartest die steht die Peter
25810. der wartende Frau Peter standet da
25811. dort drueben wartete und stehe Herr Peter
25812. das wartete die Herr Peter stehe da
25813. der Frau Peter warte dort drueben und standest
25814. dort drueben wartenden das stehe Peter
25815. da wartend der steh der Herr Peter
25816. Frau Peter warten und stehende da drueben
25817. der Peter warte standet dort drueben
25818. der Peter wartetet dort und stehende
25819. die Peter wartete dort drueben und standen
25820. da drueben warte der stehenden Herr Peter
25821. die Herr Peter wartenden da und steht
25822. dort drueben wartete und stehen Frau Peter
25823. der Frau Peter wartende stand dort drueben
25824. der Herr Peter warten stehe dort
:
57000. dort drueben wartetest und standet die Herr Peter
57001. das wartetet die Herr Peter standest da drueben
57002. das wartetest die Herr Peter stehe dort drueben
57003. die wartetest die Herr Peter standet dort drueben
57004. das wartetet die Herr Peter standest dort drueben
57005. der wartetet die Herr Peter standet da drueben
57006. das wartetest die Herr Peter standest da drueben
57007. der wartetet die Herr Peter standet dort drueben
57008. da drueben wartetet das standet die Herr Peter
57009. die Herr Peter wartetet und standet da drueben
57010. das wartetest die Herr Peter standest dort drueben
57011. der wartetest die Herr Peter standet da drueben
57012. das wartetet die Herr Peter stehst dort drueben
57013. dort drueben wartetet das standet die Herr Peter

- 57014. die Herr Peter wartetet und standet dort drueben
- 57015. der wartetest die Herr Peter standet dort drueben
- 57016. da drueben wartetest das standet die Herr Peter
- 57017. die Herr Peter wartetest und standet da drueben
- 57018. das wartetest die Herr Peter stehst dort drueben
- 57019. dort drueben wartetest das standet die Herr Peter
- 57020. die Herr Peter wartetest und standet dort drueben
- 57021. das wartetet die Herr Peter standet da drueben
- 57022. das wartetet die Herr Peter standet dort drueben
- 57023. das wartetest die Herr Peter standet da drueben
- 57024. das wartetest die Herr Peter standet dort drueben

C.2.3 Rangfolge durch Normierung der Bewertung des Microsoft Web N-gram Sprachmodells

1. dort wartet der stand der Herr Peter
2. da wartet der stand der Herr Peter
3. dort wartet der stand der Frau Peter
4. da wartet der stand der Frau Peter
5. dort wartete der stand der Herr Peter
6. dort wartet und stand der Herr Peter
7. da wartet und stand der Herr Peter
8. dort warten der stand der Herr Peter
9. dort warten und stand der Herr Peter
10. dort wartet der stand der Peter
11. da warten der stand der Herr Peter
12. da wartet der stand der Peter
13. da warten und stand der Herr Peter
14. dort wartete und stand der Herr Peter
15. dort wartete der stand der Frau Peter
16. dort wartet und stand der Frau Peter
17. dort warteten und stand der Herr Peter
18. da wartet und stand der Frau Peter
19. dort warten der stand der Frau Peter
20. dort warten und stand der Frau Peter

21. dort wartet und steht der Herr Peter
22. dort wartet das steht der Herr Peter
23. da warten der stand der Frau Peter
24. da warten und stand der Frau Peter
25. da wartet und steht der Herr Peter
- ⋮
25800. die Herr Peter wartend stand da drueben
25801. dort drueben wartetest und stehe die Peter
25802. die Herr Peter wartest da und standen
25803. da wartetest das steh der Frau Peter
25804. die Herr Peter wartende stand da
25805. die wartet Frau Peter stehen dort
25806. dort drueben wartende und stehst der Frau Peter
25807. da warte die Herr Peter und standet
25808. da drueben wartete das steh Frau Peter
25809. da drueben wartenden und stand die Peter
25810. das wartete der Herr Peter standet da
25811. Peter wartend dort drueben und stehend
25812. die Frau Peter wartetest da und steh
25813. der warte die Herr Peter stehenden dort
25814. da drueben warteten der stehst die Frau Peter
25815. der Peter warteten da und stehen
25816. der Herr Peter wartete steh da
25817. da wartetest das steht Herr Peter
25818. Frau Peter wartetest da drueben und stehen
25819. dort drueben wartest der stand die Peter
25820. der Frau Peter wartenden dort und stehend
25821. der Frau Peter wartetest da und stehe
25822. der Frau Peter wartenden und stehende dort drueben
25823. der Frau Peter wartend und standest da drueben
25824. der wartet Peter stand dort
- ⋮
57000. wartetest stehend Peter dort
57001. Peter wartende standet dort

- 57002. Peter wartest standet dort
- 57003. Peter wartetest steh dort
- 57004. Peter wartetest stehenden dort
- 57005. Peter wartetet stehende dort
- 57006. wartetet stehe Peter dort
- 57007. Peter wartetet stehend dort
- 57008. Peter wartetest stehende dort
- 57009. wartetest stehe Peter dort
- 57010. wartetet standest Peter dort
- 57011. Peter wartetest stehend dort
- 57012. wartetest standest Peter dort
- 57013. wartetet stehst Peter dort
- 57014. Peter wartetet stehe dort
- 57015. wartetest stehst Peter dort
- 57016. Peter wartetest stehe dort
- 57017. Peter wartetet standest dort
- 57018. Peter wartetest standest dort
- 57019. Peter wartetet stehst dort
- 57020. wartetet standet Peter dort
- 57021. Peter wartetest stehst dort
- 57022. wartetest standet Peter dort
- 57023. Peter wartetet standet dort
- 57024. Peter wartetest standet dort

Anhang D

Nicht öffentlich zugängliche Quellen

D.1 Korrespondenz mit Prof. Horst Ebbinghaus [Ebb15]

Von: Horst Ebbinghaus
 An: Jens Lindeman <lindemann@b-tu.de >
 Betreff: Re: Anfrage Glossennotaion DGS Beispielsätze
 Gesendet: Mo 27.04.2015 19:53

Re: Anfrage Glossennotaion DGS Beispielsätze

Sehr geehrter Herr Lindemann,

eine verbindliche Beantwortung Ihrer Fragen ist nicht ganz einfach. Dies liegt zum einen natürlich an dem Modalitätsunterschied, der sich bei einer rein schriftlichen Darstellung nicht überbrücken lässt. Es gibt aber auch prinzipielle strukturelle Gründe:

(1) Es ist zumindest umstritten, ob wir bei der DGS im gleichen Sinne von kategorialen ‚Wortarten‘ sprechen können wie bei Lautsprachen. Mit Sicherheit gibt es keine Partizipien und keine regulären morphologischen Prozesse, um aus Verben Adjektive oder andere attributartige Ausdrücke abzuleiten.

(2) Wir können zwischen Deutsch und DGS auch nicht unbedingt von einer Entsprechung der Syntagmen unterhalb der Satzebene (Satzglieder) ausgehen. Deshalb gibt es bei den Differenzierungen zwischen Ihren deutschen Beispielen meist keine unmittelbaren syntaktischen Entsprechungen in der DGS. Vielmehr lassen sich nur semantische Interpretationen der syntaktischen Unterschiede der deutschen Beispiele in der DGS nachvollziehen, indem jeweils bestimmte Bedeutungsaspekte (entweder stehen oder warten) fokussiert werden, was (abgesehen von Stellungsunterschieden) in der Regel durch zusätzliche nonmanuelle Mittel geschieht. Dabei gibt es oft sogar mehrere Möglichkeiten, wobei dann jeweils unterschiedliche Kontexte mitzudenken sind. Insofern lassen sich die Prädikate stehen und warten durchaus in vielfältiger Weise dem Argument Peter zuordnen, nur entsprechen diese nicht 1:1 den semantischen Unterschieden zwischen den deutschen Beispielen.

(3) Die Beispiele, die ich dabei mit meinem muttersprachlichen Mitarbeiter durchgespielt habe, führen immer zu einer Zweiteilung des Satzes in zwei letztlich unabhängige Propositionen (PETER WARTEN. PETER STEHEN. etc), was m. E. damit zusammenhängt, dass gebärdensprachliche Äußerungen sich sinnvoll und einigermaßen zuverlässig ohnehin nur nach ihrem propositionalen Gehalt gliedern lassen. Oder, wenn Sie so wollen, gebärdensprachliche Äußerungen sind Ketten von kurzen ‚Sätzen‘, die jeweils aus Argument und Prädikat (im semantischen bzw. logischen Sinn) bestehen. D. h., für einen syntaktisch fundierten Satzbegriff der DGS können wir keine allgemeingültigen Kriterien angeben. Daraus ergibt sich auch die Antwort auf Ihre letzte Frage unter (1): Die Variationsmöglichkeit innerhalb eines ‚Satzes‘ ist in der Tat stark beschränkt, nicht aber die Möglichkeit, verschiedene semantische Inhalte durch eine Abfolge von unterschiedlichen ‚Elementarpropositionen‘ auszudrücken.

Zu Ihren Glossenbeispielen hätten wir noch anzumerken, dass DA nur in sehr speziellen Kontexten, in denen Existenz bzw. Anwesenheit (im Ggs. zu Abwesenheit) betont würde, verwendet werden kann. Im unmarkierten Fall, der in Ihren Beispielen gegeben zu sein scheint, würde immer ein INDEX benutzt.

Ich bin mir bewusst, dass meine Antworten wegen Ihrer Abstraktheit für Sie unbefriedigend bleiben müssen. Welche unterschiedlichen gebärdensprachlichen Realisierungsmöglichkeiten für Ihre Beispiele in Betracht kommen, ist mit einer einfachen Glossennotation nicht einzufangen und ließe sich nur anschaulich demonstrieren. Ich hoffe trotzdem, dass Ihnen meine prinzipiellen Anmerkungen eine gewisse Orientierung bieten.

Mit besten Grüßen

Horst Ebbinghaus

Am 14.04.2015 um 14:55 schrieb Jens Lindemann <lindemann@b-tu.de >:

Sehr geehrter Prof. Dr. Ebbinghaus,

im Rahmen meiner Promotion an der BTU Cottbus-Senftenberg beschäftige ich mich u. a. mit der semantischen Interpretation von erkannten natürlichsprachlichen Äußerungen (Laut- und Gebärdensprache) durch technische Systeme. In meinem Theorieteil habe ich Beispielsätze aus einer von mir betreuten Masterarbeit aus dem Jahr 2010 aufgegriffen. Die vorgegeben Sätze wurden von einer Gebärdensprachdolmetscherin gebärdet. Im Anschluss wurden die Videoaufnahmen mittels Glossennotation (nach: Papaspyrou, Chrissostomos et al: „Grammatik der deutschen Gebärdensprache aus der Sicht gehörloser Fachleute.“ Signum, 2008.) transkribiert.

Da ich meine Kenntnisse zur Deutschen Gebärdensprache lediglich selbst angeeignet habe, benötige ich als Ingenieur für eine mögliche Veröffentlichung auch die Meinung/Unterstützung eines Linguisten auf dem Gebiet der Gebärdensprache. Ich hoffe Sie oder einer ihrer Mitarbeiter/Kollegen können mir zu dem gewählten Theoriebeispiel Auskunft erteilen.

Zu der Aussage „Dort drüben steht Peter und wartet.“ sind verschiedene Satzvariationen erstellt worden, die ich nun genauer untersuche. Die Beispielsätze finden Sie im pdf-Dokument im Anhang dieser E-Mail. Vier mögliche Notationen von möglichen Gebärdensprachrealisierungen liegen mir vor (vgl. Beispielsätze 1 – 4 im Anhang), wobei die Gebärdensprachdolmetscherin den Sachverhalt des „stehen“ – sich also aufrecht befinden (anders als z. B. sitzen) in den Transkriptionen zu den Beispielsätzen 3 und 4 wohl als unwichtig ansah und deshalb nicht artikulierte. Daraus ergibt sich z. B. das Problem, dass derzeit die semantische Analyse des Satzes 3 „Peter steht dort drüben und wartet.“ dieselbe Bedeutungsstruktur ergibt, wie die Aussagen „Peter ist/befindet sich dort drüben und wartet.“ oder sogar „Peter sitzt dort drüben und wartet.“ (wenn sitzen auch vernachlässigt wird).

Frage 1: Wie sieht die Transkription der Sätze 3 und 4 aus, wenn der Aspekt des „Stehens“ noch berücksichtigt würde. Ist die Index-Gebärde hier noch sinnvoll oder wird dies auch durch die Gebärde „DA“ ausgedrückt?

Die Sätze 5 und 6 wurden nicht gebärdet, da die Dolmetscherin diese identisch übersetzen würde bzw. keine passendere alternative Ausdrucksmöglichkeit hatte. Die Sätze 7 bis 11 für die geschriebene deutsche Sprache habe ich jetzt noch ergänzt, da dies auch mögliche Realisierungen der geschriebenen deutschen Sprache wären. Durch die Topikalisierung ergeben sich zwar auch für die DGS verschiedene syntaktische Strukturen, allerdings scheint mir die Variationsmöglichkeit hier stärker begrenzt, als bei der deutschen Lautsprache.

Frage 2: Gibt es in der Deutschen Gebärdensprache die Möglichkeit die verwendete Partizipien (in den Sätzen 4-11) zu artikulieren oder wird dies immer durch die 2 verschiedenen Handlungen (hier die Verben: stehen und warten) ausgedrückt? Möglicherweise gibt es ja eine Variation in der DGS die dieses Partizip ausdrückt. Vielleicht ist die praktische Verwendung im Sprachgebrauch auch unüblich, aber theoretisch möglich. Können Sie mir hierzu Auskunft geben und ggf. mögliche Notationen mitteilen! Vielen Dank

Erläuterung: In der geschriebenen deutschen Sprache wird das Partizip (hier: stehend, wartend) verwendet. Durch das Anhängen eines „-d“ an die Grundform des Verbes erhält die Verbform die Bedeutung eines Adjektivs. Sätze 4 + 5: wartend beschreibt hier das Stehen von Peter genauer Sätze 6 +7: wartend beschreibt hier eine Eigenschaft von Peter (wartende Peter) Sätze 8 + 9: stehend - beschreibt hier die Eigenschaft von Peter (stehende Peter) Sätze 10 + 11: stehend beschreibt hier das Warten von Peter genauer

Ich wäre Ihnen sehr dankbar, wenn Sie mir bei dieser Problematik weiterhelfen, bzw. vermittelnd behilflich sein könnten. Für Rückfragen stehe ich ihnen selbstverständlich gerne zur Verfügung.

Mit freundlichen Grüßen

Jens Lindemann

D.2 Korrespondenz mit Prof. Rachel Rosenstock [Ros15]

Von: Rachel Rosenstock
An: Jens Lindeman <lindemann@b-tu.de >
Betreff: Re: Glossennotation "PETER"
Gesendet: Mi 16.09.2015 10:01

Lieber Herr Lindemann, es war das jüdische Neujahrsfest und ich daher einige Tage offline. Nun meine Antworten unten zwischen Ihrer Mail:

1. Der PETER ist für die Gebärdensprachdarstellung durch MÄDCHEN ersetzt worden - kann ich unter der Annahme einer bekannten Namensgebärde für den gemeinten Peter diese Substitution wieder rückgängig machen? Oder gibt es dahingehend Einschränkungen? Hier reicht das einfache 'ja'!

2. Im Beispiel 6 (Hände: MÄDCHEN STEHEN...DORT...STEHEN) kommt das STEHEN 2x vor, hingegen das WARTEN nicht. Ist dies so korrekt? Wenn ja, welche Bedeutung (Betonung oder grammatische Information) hat diese Reduplikation in diesem Fall? Leider ist ja die DGS-Grammatik noch nicht wirklich umfassend beschrieben. Am ausführlichsten behandeln m.E. Happ/Vorköper in dem sog. Frankfurter Material diese Frage. Meine Intuition sagt mir hier es handelt sich um eine Hervorhebung des Konzepts 'stehen', aber das lässt sich abschließend nicht wirklich sagen. Diese grammatischen Permutationen sind nach meinem Wissen noch nicht erforscht.

3. Das verwendete Relativpronomen SELBST ist in diesem Fall auf die 3. Person Singular bezogen? Dies müsste dann etwa dieser Variante (mit G-Handform) von wikisign entsprechen -> <http://dgs.wikisign.org/wiki/Selbst> Ja, genau. Die Gebärde ist ortsbezogen und kann somit variieren (siehe Antworten bei 4. A)

4. A) Ist es im Prinzip möglich bei der Gebärde DORT mit dem Zeigefinger auf jeden beliebigen Punkt im Gebärdenraum zu referenzieren (G-Handform?)? Ja, das ist es.

B1) Kann die Gebärde DORT in diesem Fall auch als eine genauer spezifizierte (bzw. interpretierte) INDEX-Gebärde betrachtet werden? Allgemein findet man oft in Notationen die unspezifizierte Angabe einer INDEX-Gebärde. Frau Hausmann hatte diese in ihren Beispielsätzen so verwendet und transkribiert. Ja und nein - die oft als INDEX verglossete Gebärde erfüllt in DGS vielen Funktionen, wie Sie selber unter B2) schreiben. Als neutrale Lokaldeixis könnte DORT durchaus verwechselt werden, allerdings gibt es mimische Markierungen und das Mundbild, welche eine Differenzierung zulassen. Dies gilt z.B. auch für das Demonstrativpronomen (oft mit einer anderen Handflächenorientierung, zusammengezogenen Augen und auseinander gezogenen Mundwinkeln markiert). Ich habe den Text nicht präsent, aber ich meine, Papaspyrou et al. 2007 hätten hierzu ausführlicher beschrieben (allerdings auch nur anekdotisch und nicht empirisch belegt).

B2) Wäre es somit möglich die Gebärde für die Lokaldeixis "DORT" mit einem Artikel, Demonstrativpronomen oder Personalpronomen zu verwechseln, wenn diese als Einzelgebärde artikuliert / betrachtet wird?

5. Die Gebärden wurden alle im neutralen Gebärdenraum, also vor Sprecher(in) ausgeführt? Frau Steinbach hat einen bestimmten Punkt vor dem Oberkörper gewählt, auf den alle ortsbezogenen Gebärden sich beziehen. DORT, SELBST und STEHEN beziehen sich auf diesen Punkt. Ohne spezifischen Kontext ist dies die bedeutungsneutrale Variante. Im natürlichen Diskurs würde dies aber so nicht gebärdet.

Ich hoffe, das hilft weiter! Sollten Sie Zugang zu den Videodaten haben wollen, um die Form der Gebärden zu vergleichen, melden Sie sich gern.

Mit bestem Gruß,

Rachel Rosenstock

Literaturverzeichnis

- [ARS⁺07] Allauzen, C., Riley, M., Schalkwyk, J., Skut, W. und Mohri, M.; *OpenFst: A general and efficient weighted finite-state transducer library. Implementation and Application of Automata*, 2007.
- [BB90] Boyes-Braem, P.; *Einführung in die Gebärdensprache und ihre Erforschung*; Signum, Hamburg, 1990.
- [BBC⁺09] Barrachina, S., Bender, O., Casacuberta, F., Civera, J., Cubel, E., Khadivi, S., Lagarda, A., Ney, H., Tomás, J., Vidal, E. und Vilar, J.M.; *Statistical Approaches to Computer-Assisted Translation*; Computational Linguistics 35 (1), S. 3–28, 2009.
- [BDD98] Bernsen, N., Dybkjær, L. und Dybkjær, H.; *Wizard of Oz Simulation*; in: *Designing Interactive Speech Systems*; S. 127–160; Springer London, 1998.
- [Bie08] Bierwisch, M.; *Die Entwicklung des Gehirns und der Sprache*; in: J. Klose und J. Oehler, Hrsg., *Gott oder Darwin?*; S. 173–200; Springer Berlin Heidelberg, 2008.
- [Bil88] Biljon, W.R.v.; *Extending Petri nets for specifying man-machine dialogues*; International Journal of Man-Machine Studies 28 (4), S. 437–455, 1988.
- [BK03] Bilmes, J.A. und Kirchhoff, K.; *Factored Language Models and Generalized Parallel Backoff*; in: *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology: Companion Volume of the Proceedings of HLT-NAACL 2003-short Papers - Volume 2*; NAACL-Short '03; S. 4–6; Association for Computational Linguistics, 2003.
- [BKPZ06] Beecken, A., Keller, J., Prillwitz, S. und Zienert, H.; *Grundkurs Deutsche Gebärdensprache, Stufe 1; Arbeitsbuch*; Bd. 3 von *Gebärdensprachlehre*; Signum-Verl., Seedorf, 2006.
- [BLV⁺06] Benedí, J.m., Lleida, E., Varona, A., Castro, M.j., Galiano, I., Justo, R., Iñigo López De Letona und Miguel, A.; *Design and acquisition of a telephone spontaneous speech dialogue corpus in spanish: Dihana*; in: *Fifth International Conference on Language Resources and Evaluation (LREC)*; S. 1636–1639, 2006.

- [BMB⁺06] Bonneau-Maynard, H., Ayache, C., Bechet, F., Denis, A., Kuhn, A., Lefevre, F., Mostefa, D., Quignard, M., Rosset, S., Servan, C. und Villaneau, J.; *Results of the french evalda-media evaluation campaign for literal understanding*; in: *In 5th International Conference on Language Resources and Evaluation (LREC2006)*, 2006.
- [BMON03] Bender, O., Macherey, K., Och, F.J. und Ney, H.; *Comparison of Alignment Templates and Maximum Entropy Models for Natural Language Understanding*; in: *Proceedings of the Tenth Conference on European Chapter of the Association for Computational Linguistics - Volume 1*; S. 11–18; Association for Computational Linguistics, Stroudsburg, 2003.
- [BMQD09] Bonneau-Maynard, H., Quignard, M. und Denis, A.; *MEDIA: a semantically annotated corpus of task oriented dialogs in French*; *Language Resources and Evaluation* 43 (4), S. 329–354, 2009.
- [BPP96] Berger, A.L., Pietra, V.J.D. und Pietra, S.A.D.; *A Maximum Entropy Approach to Natural Language Processing*; *Comput. Linguist.* 22 (1), S. 39–71, 1996.
- [BPPM93] Brown, P.F., Pietra, V.J.D., Pietra, S.A.D. und Mercer, R.L.; *The Mathematics of Statistical Machine Translation: Parameter Estimation*; *Comput. Linguist.* 19 (2), S. 263–311, 1993.
- [BR00] Bangalore, S. und Rambow, O.; *Exploiting a probabilistic hierarchical model for generation*; in: *Proceedings of the 18th conference on Computational linguistics-Volume 1*; S. 42–48, 2000.
- [BSH08] Benesty, J., Sondhi, M.M. und Huang, Y.; *Springer handbook of speech processing*; Springer, Berlin and London, 2008.
- [Bus03] Busemann, S.; *Generierung natürlchsprachlicher Texte*; in: G. Görz, C.R. Rollinger und J. Schneeberger, Hrsg., *Handbuch der Künstlichen Intelligenz*; S. 783–814; Oldenbourg Wissenschaftsverlag Verlag, München, 2003.
- [Cal15] Calvo, M.; *A Strategy for Multilingual Spoken Language Understanding based on Graphs of Linguistic Units*; Dissertation; Universitat Politècnica de València, 2015.
- [CEE10] Carstensen, K.U., Ebert, C. und Ebert, C.; *Computerlinguistik und Sprachtechnologie: Eine Einführung*; Spektrum Akademischer Verlag, Heidelberg, 2010.
- [CG99] Chen, S.F. und Goodman, J.; *An empirical study of smoothing techniques for language modeling*; *Computer Speech & Language* 13 (4), S. 359–394, 1999.
- [CGS13] Calvo, M., Garcénez, F. und Sanchis, E.; *Exploiting multiple hypotheses for Multilingual Spoken Language Understanding*; in: J. Hockenmaier und S. Riedel, Hrsg., *Proceedings of the Seventeenth Conference on Computational Natural Language Learning, CoNLL 2013*; S. 193–201; ACL, 2013.

- [CHB11] Cooper, H., Holt, B. und Bowden, R.; *Sign Language Recognition*; in: T.B. Moeslund, A. Hilton, V. Krüger und L. Sigal, Hrsg., *Visual Analysis of Humans*; S. 539–562; Springer London, 2011.
- [Che76] Chen, P.P.S.; *The entity-relationship model—toward a unified view of data*; ACM Transactions on Database Systems (TODS) 1 (1), S. 9–36, 1976.
- [CHG+16] Calvo, M., Hurtado, L.F., Garcia, F., Sanchis, E. und Segarra, E.; *Multilingual Spoken Language Understanding using graphs and multiple translations*; Computer Speech & Language 38, S. 86–103, 2016.
- [CHGS12] Calvo, M., Hurtado, L.F., García, F. und Sanchís, E.; *A Multilingual SLU System Based on Semantic Decoding of Graphs of Words*; in: *Advances in Speech and Language Technologies for Iberian Languages*; Bd. 328 von *Communications in Computer and Information Science*; S. 158–167; Springer Berlin Heidelberg, 2012.
- [CLM+01] Casacuberta, F., Llorens, D., Martinez, C., Molau, S., Nevado, F., Ney, H., Pastor, M., Pico, D., Sanchis, A., Vidal, E. und Vilar, J.; *Speech-to-speech translation based on finite-state transducers*; in: *Acoustics, Speech, and Signal Processing, 2001. Proceedings. (ICASSP '01). 2001 IEEE International Conference on*; Bd. 1; S. 613–616 vol.1, 2001.
- [CM79] Christaller, T. und Metzger, D.; *Augmented Transition Network: Grammatiken : "Aspekte des ATN-Formalismus in sprachverarbeitenden Systemen"*; Bd. 1 ; 1-2 von *Künstliche Intelligenz*, 1979.
- [CM05] Carreras, X. und Màrquez, L.; *Introduction to the CoNLL-2005 Shared Task: Semantic Role Labeling*; in: *Proceedings of the Ninth Conference on Computational Natural Language Learning*; CONLL '05; S. 152–164; Association for Computational Linguistics, Stroudsburg, 2005.
- [CM06] Crystal Nakatsu und Michael White; *Learning to say it well: Reranking realizations by predicted synthesis quality*; in: *In Proceedings of COLING-ACL 2006*; S. 1113–1120, 2006.
- [CV95] Cortes, C. und Vapnik, V.; *Support-Vector Networks*; Machine Learning 20 (3), S. 273–297, 1995.
- [Dah87] Dahlberg, I.; *Die gegenstandsbezogene, analytische Begriffstheorie und ihre Definitionsarten*; in: B. Ganter, R. Wille und K. Wolff, Hrsg., *Beiträge zur Begriffsanalyse*; S. 9–22; BI Wissenschaftsverlag, Mannheim and Wien and Zürich, 1987.
- [DES04] DESIRE; *Aachener Glossenumschrift: Technical report*, 2004.
- [DH95] Dotter, F. und Holzinger, D.; *Typologie und Gebärdensprache: Sequentialität und Simultanität*; Sprachtypologie und Universalienforschung (48), S. 311–349, 1995.

- [Din10] Dinarelli, M.; *Spoken Language Understanding: from Spoken Utterances to Semantic Structures*; Dissertation; Università degli Studi di Trento, 2010.
- [DJA93] Dahlbäck, N., Jönsson, A. und Ahrenberg, L.; *Wizard of Oz studies: why and how*; in: W.D. Gray, D. Murray und W.E. Hefley, Hrsg., *Proceedings of the 1st international conference on Intelligent user interfaces*; S. 193–200; ACM Press, New York, 1993.
- [Dot01] Dotter, F.; *Gebärdensprachforschung*; in: M. Haspelmath, Hrsg., *Language typology and language universals*; Bd. 20 von *Handbooks of linguistics and communication science Handbücher zur Sprach- und Kommunikationswissenschaft*; S. 141–154; W. de Gruyter, Berlin and New York, 2001.
- [DSD+08] Dreuw, P., Stein, D., Deselaers, T., Rybach, D., Zahedi, M., Bungeroth, J. und H., N.; *Spoken language processing techniques for sign language recognition and translation*; *Technology and Disability* 20 (2), S. 121–133, 2008.
- [DSTHT12] Deoras, A., Sarikaya, R., Tur, G. und Hakkani-Tur, D.; *Joint Decoding for Speech Recognition and Semantic Tagging*; in: *INTERSPEECH*; S. 1067–1070, 2012.
- [Ebb15] Ebbinghaus, Horst; *Anfrage Glossennotation DGS Beispielsätze [E-Mail]*, 27.04.2015.
- [EHH12] Eichmann, H., Hansen, M. und Heßmann, J.; *Handbuch deutsche Gebärdensprache: Sprachwissenschaftliche und anwendungsbezogene Perspektiven*; Bd. 50 von *Internationale Arbeiten zur Gebärdensprache und Kommunikation Gehörloser*; Signum, Seedorf, 2012.
- [Elm90] Elman, J.L.; *Finding Structure in Time*; *Cognitive Science* 14 (2), S. 179–211, 1990.
- [EPR+96] Epstein, M., Papineni, K., Roukos, S., Ward, T. und Della Pietra, S.; *Statistical natural language understanding using hidden clumpings*; in: *Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings., 1996 IEEE International Conference on*; Bd. 1; S. 176–179 vol. 1, 1996.
- [ES05] Erik Velldal und Stephan Oepen; *Maximum Entropy Models for Realization Ranking*; in: *In Proceedings of the 10th Machine Translation Summit (pp. 109)*; S. 116, 2005.
- [Fel12] Fellbaum, K.; *Sprachverarbeitung und Sprachübertragung*; Springer, Berlin and Heidelberg; 2. Aufl., 2012.
- [FG91] Fraser, N.M. und Gilbert, G.N.; *Simulating speech systems*; *Computer Speech & Language* 5 (1), S. 81–99, 1991.
- [Fin14] Fink, G.A.; *Markov models for pattern recognition: From theory to applications*; *Advances in computer vision and pattern recognition*; Springer-Verlag London; 2. Aufl., 2014.

- [FKM09] Fichtner, I., Kuske, D. und Meinecke, I.; *Traces, series-parallel posets, and pictures: A weighted study*; in: M. Droste, W. Kuich und H. Vogler, Hrsg., *Handbook of Weighted Automata*; Monographs in Theoretical Computer Science. An EATCS Series; S. 405–450; Springer Berlin Heidelberg, 2009.
- [Fus01] Fuster, J.M.; *The Prefrontal Cortex—An Update: Time Is of the Essence*; *Neuron* 30 (2), S. 319–333, 2001.
- [FW08] Francois Mairesse und Walker, M.A.; *Trainable Generation of Big-Five Personality Styles through Data-driven Parameter Estimation*; 46th Annual Meeting of the Association for Computational Linguistics , 2008.
- [GFG+95] Glass, J., Flammia, G., Goodine, D., Phillips, M., Polifroni, J., Sakai, S., Seneff, S. und Zue, V.; *Multilingual Spoken-language Understanding in the MIT Voyager System*; *Speech Commun* 17 (1-2), S. 1–18, 1995.
- [GJ02] Gildea, D. und Jurafsky, D.; *Automatic Labeling of Semantic Roles*; *Computational linguistics* 28 (3), S. 245–288, 2002.
- [GNL+10] Gao, J., Nguyen, P., Li, X., Thrasher, C., Li, M. und Wang, K.; *A Comparative Study of Bing Web N-gram Language Models for Web Search and Natural Language Processing*; in: *Proceeding of the 33rd Annual ACM SIGIR Conference*; Association for Computing Machinery, Inc, 2010.
- [GRW97] Gorin, A., Riccardi, G. und Wright, J.; *How may I help you?*; *Speech Communication* 23 (1-2), S. 113–127, 1997.
- [GWG11] Guo, Y., Wang, H. und van Genabith, J.; *Dependency-based n-gram models for general purpose sentence realisation*; *Natural Language Engineering* 17 (04), S. 455–483, 2011.
- [GWH85] Green, P. und Wei-Haas, L.; *The rapid development of user interfaces: Experience with the Wizard of Oz method*; in: *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*; Bd. 29; S. 470–474, 1985.
- [HDR+11] Hahn, S., Dinarelli, M., Raymond, C., Lefevre, F., Lehnen, P., Mori, R.d., Moschitti, A., Ney, H. und Riccardi, G.; *Comparing Stochastic Approaches to Spoken Language Understanding in Multiple Languages*; *Audio, Speech, and Language Processing, IEEE Transactions on* 19 (6), S. 1569–1583, 2011.
- [Heß01] Heßmann, J.; *Gehörlos so! Materialien zur Gebärdensprache*; Bd. 32 von *Internationale Arbeiten zur Gebärdensprache und Kommunikation Gehörloser*; Signum, Hamburg, 2001.
- [HKL+09] Huber, M., Kölbl, C., Lorenz, R., Römer, R. und Wirsching, G.; *Semantische Dialogmodellierung mit gewichteten Merkmal-Werte-Relationen*; in: R. Hoffmann, Hrsg., *Elektronische Sprachsignalverarbeitung 2009*; Bd. 53 von *Studenten zur Sprachkommunikation*; TUDpress, Dresden, 2009.

- [HKLW08] Huber, M., Kölbl, C., Lorenz, R. und Wirsching, G.; *Ein Petrinetz-Modell zur Informationsübertragung per Dialog*; in: N. Lohmann und W. Karsten, Hrsg., *5th German Workshop on Algorithms and Tools for Petri Nets*, 2008.
- [HKLW13] Huber, M., Kölbl, C., Lorenz, R. und Wirsching, G.; *Konstruktion von UMP-Transduktoren aus Wizard-of-Oz Daten*; in: P. Wagner, Hrsg., *Elektronische Sprachsignalverarbeitung 2013*; Bd. 65 von *Studientexte zur Sprachkommunikation*; S. S. 111 – 118; TUDpress, Dresden, 2013.
- [HMW12] Hinzen, W., Machery, E. und Werning, M., Hrsg.; *The Oxford Handbook of Compositionality*; Oxford University Press, 2012.
- [Hör77] Hörmann, H.; *Psychologie der Sprache*; Kurzlehrbücher Psychologie; Springer-Verlag, Berlin and New York; 2. Aufl., 1977.
- [HV06] Happ, D. und Vorköper, M.O.; *Deutsche Gebärdensprache: Ein Lehr- und Arbeitsbuch*; Fachhochsch.-Verl, Frankfurt am Main, 2006.
- [HW15] Hoffmann, R. und Wolff, M.; *Intelligente Signalverarbeitung 2: Signalerkennung*; Springer Vieweg, 2015.
- [HY05] He, Y. und Young, S.; *Semantic processing using the Hidden Vector State model*; *Computer Speech & Language* 19 (1), S. 85–106, 2005.
- [IBO06] Isard, A., Brockmann, C. und Oberlander, J.; *Individuality and Alignment in Generated Dialogues*; in: *Proceedings of the Fourth International Natural Language Generation Conference*; S. 25–32; Association for Computational Linguistics, Stroudsburg, 2006.
- [ISO00] *ISO 1087-1: Vocabulary*; ISO, Genève, 2000.
- [JBL13] Jabaian, B., Besacier, L. und Lefevre, F.; *Comparison and Combination of Lightly Supervised Approaches for Language Portability of a Spoken Language Understanding System*; *Audio, Speech, and Language Processing*, *IEEE Transactions on* 21 (3), S. 636–648, 2013.
- [JHJ98] Johnson, M.T., Harper, M.P. und Jamieson, L.H.; *Interfacing Acoustic Models With Natural Language Processing Systems*; in: *In Proc. ICSLP '98*; S. 2419–2422, 1998.
- [JL08] Jeong, M. und Lee, G.G.; *Triangular-Chain Conditional Random Fields*; *IEEE Transactions on Audio, Speech, and Language Processing* 16 (7), S. 1287–1302, 2008.
- [JM00] Jurafsky, D. und Martin, J.H.; *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition*; Prentice Hall series in artificial intelligence; Prentice Hall, Upper Saddle River and N.J., 2000.
- [JM14] Jurafsky, D. und Martin, J.H.; *Speech and language processing*; Always learning; Pearson Education, Harlow; 2. Aufl., 2014.

- [Kar12] Karnagel, K.; *Semantische Modellierung von System-Benutzer-Interaktionen am Beispiel einer Home-Entertainment-Anwendung*; Masterarbeit; BTU Cottbus, 2012.
- [Kel83] Kelley, J.F.; *An empirical methodology for writing user-friendly natural language computer applications*; in: *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*; S. 193–196, 1983.
- [Kel84] Kelley, J.F.; *An iterative design methodology for user-friendly natural language office information applications*; *ACM Transactions on Information Systems* 2 (1), S. 26–41, 1984.
- [KFL01] Kschischang, F.R., Frey, B.J. und Loeliger, H.A.; *Factor graphs and the sum-product algorithm*; *IEEE Transactions on Information Theory* 47 (2), S. 498–519, 2001.
- [KFN15] Koller, O., Forster, J. und Ney, H.; *Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers*; *Computer Vision and Image Understanding* 141, S. 108–125, 2015.
- [KH95] Knight, K. und Hatzivassiloglou, V.; *Two-level, Many-paths Generation*; in: *Proceedings of the 33rd Annual Meeting on Association for Computational Linguistics*; S. 252–260; Association for Computational Linguistics, Stroudsburg, 1995.
- [KLC98] Kaelbling, L.P., Littman, M.L. und Cassandra, A.R.; *Planning and Acting in Partially Observable Stochastic Domains*; *Artif. Intell.* 101 (1-2), S. 99–134, 1998.
- [Klo11] Klos, V.; *Komposition und Kompositionalität, Möglichkeiten und Grenzen der semantischen Dekodierung von Substantivkomposita*; de Gruyter, Berlin und Boston, 2011.
- [KM01] Kudo, T. und Matsumoto, Y.; *Chunking with Support Vector Machines*; in: *Proceedings of the Second Meeting of the North American Chapter of the Association for Computational Linguistics on Language Technologies*; S. 1–8; Association for Computational Linguistics, Stroudsburg, 2001.
- [KM09] Kanis, J. und Müller, L.; *Advances in Czech - Signed Speech Translation*; Bd. 5729 von *Lecture Notes in Computer Science*; S. 48–55; Springer, 2009.
- [KN06] Khoo, C.S.G. und Na, J.C.; *Semantic relations in information science*; *Annual Review of Information Science and Technology* 40(1), S. 157–228, 2006.
- [KNH11] Kipp, M., Nguyen, Q. und Heloir, A.; *Machbarkeitsstudie zur Abschätzung der Nutzungsmöglichkeiten von Gebärdensprache-avataren*; Bd. 417 von *Forschungsbericht*; Bundesministerium für Arbeit und Soziales, Bonn, 2011.
- [Koe05] Koehn, P.; *Europarl: A parallel corpus for statistical machine translation*; in: *MT summit*; Bd. 5; S. 79–86, 2005.

- [KRRL12] Karnagel, K., Römer, R., Rogge, S. und Lindemann, J.; *Semantische Modellierung von System-Benutzer-Interaktionen am Beispiel einer Home-Entertainment Steuerung*; in: M. Wolff, Hrsg., *Elektronische Sprachsignalverarbeitung 2012*; Bd. 64 von *Studientexte zur Sprachkommunikation*; S. 306–313; TUDpress, Dresden, 2012.
- [KW11] Kölbl, C. und Wirsching, G.; *Language Modeling with Utterance-Meaning Pairs*; Bd. 2011,12 von *Report / Institut für Informatik*; Inst. für Informatik, Augsburg, 2011.
- [KWL⁺14] Klimczak, P., Wolff, M., Lindemann, J., Petersen, C. und Zooglauer, T.; *Die kognitive Heizung*; in: R. Hoffmann, Hrsg., *Elektronische Sprachsignalverarbeitung 2014*; Bd. 71 von *Studientexte zur Sprachkommunikation*; TUDpress, Dresden, 2014.
- [Lan00] Langkilde, I.; *Forest-based Statistical Sentence Generation*; in: *Proceedings of the 1st North American Chapter of the Association for Computational Linguistics Conference*; S. 170–177; Association for Computational Linguistics, Stroudsburg, 2000.
- [Lef06] Lefevre, F.; *A DBN-based multi-level stochastic spoken language understanding system*; in: *Spoken Language Technology Workshop, 2006. IEEE*; S. 78–81, 2006.
- [LH12] Lorenz, R. und Huber, M.; *Petri Net Transducers in Semantic Dialogue Modelling*; in: M. Wolff, Hrsg., *Elektronische Sprachsignalverarbeitung 2012*; Bd. 64 von *Studientexte zur Sprachkommunikation*; S. 298–305; TUDpress, Dresden, 2012.
- [LH13] Lorenz, R. und Huber, M.; *Realising the translation of utterances into meanings by Petri Net Transducers*; in: P. Wagner, Hrsg., *Elektronische Sprachsignalverarbeitung 2013*; Bd. 65 von *Studientexte zur Sprachkommunikation*; S. 103–110; TUDpress, Dresden, 2013.
- [LHW14] Lorenz, R., Huber, M. und Wirsching, G.; *On Weighted Petri Net Transducers*; in: *Application and Theory of Petri Nets and Concurrency*; S. 233–252; Springer, 2014.
- [Lin12] Lindemann, J.; *Vergleich der Produktionsgeschwindigkeit von Laut- und Gebärdensprache*; in: M. Wolff, Hrsg., *Elektronische Sprachsignalverarbeitung 2012*; Bd. 64 von *Studientexte zur Sprachkommunikation*; S. 236–243; TUDpress, Dresden, 2012.
- [Lin14] Lindemann, J.; *Semantische Verarbeitung von Gebärdensprache in intelligenten hierarchischen Sprachdialogsystemen*; in: R. Hoffmann, Hrsg., *Elektronische Sprachsignalverarbeitung 2014*; Bd. 71 von *Studientexte zur Sprachkommunikation*; S. 118–125; TUDpress, Dresden, 2014.

- [Lin15] Lindemann, J.; *Interpretation von Gebärdenspracheingaben am Beispiel einer Home-Entertainment-Steuerung*; in: G. Wirsching, Hrsg., *Elektronische Sprachsignalverarbeitung 2015*; Bd. 78 von *Studententexte zur Sprachkommunikation*; TUDpress, Dresden, 2015.
- [LK98] Langkilde, I. und Knight, K.; *Generation That Exploits Corpus-based Statistical Knowledge*; in: *Proceedings of the 17th International Conference on Computational Linguistics - Volume 1*; S. 704–710; Association for Computational Linguistics, Stroudsburg, 1998.
- [LMP01] Lafferty, J.D., McCallum, A. und Pereira, F.C.N.; *Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data*; in: *Proceedings of the Eighteenth International Conference on Machine Learning*; ICML '01; S. 282–289; Morgan Kaufmann Publishers Inc, San Francisco and CA and USA, 2001.
- [LNL09] Lu, W., Ng, H.T. und Lee, W.S.; *Natural Language Generation with Tree Conditional Random Fields*; in: *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1 - Volume 1*; S. 400–409; Association for Computational Linguistics, Stroudsburg, 2009.
- [LNP08] Linke, A., Nussbaumer, M. und Portmann-Tselikas; *Studienbuch Linguistik*; Bd. 121 von *Germanistische Linguistik*; Niemeyer, Tübingen; 4. Aufl., 2001.
- [LP95] Levin, E. und Pieraccini, R.; *Concept-based spontaneous speech understanding system*; in: *Fourth European Conference on Speech (EUROSPEECH '95)*; S. 555–558, 1995.
- [LPE00] Levin, E., Pieraccini, R. und Eckert, W.; *A stochastic model of human-machine interaction for learning dialog strategies*; *IEEE Transactions on Speech and Audio Processing* 8 (1), S. 11–23, 2000.
- [LS92] Lefebvre, D. und Saridis, G.; *A computer architecture for intelligent machines*; in: *Robotics and Automation, 1992. Proceedings., 1992 IEEE International Conference on*; S. 2745–2750 vol.3, 1992.
- [Lun12] Lunze, J.; *Ereignisdiskrete Systeme: Modellierung und Analyse dynamischer Systeme mit Automaten, Markovketten und Petrinetzen*; Oldenbourg, München; 2. Aufl., 2012.
- [Mac09] Macherey, K.; *Statistical methods in natural language understanding and spoken dialogue systems*; Dissertation; RWTH Aachen, 2009.
- [Mau06] Mauser, S.; *Semantiken von Petrinetzen - ein algebraischer Ansatz, der zwischen nebenläufigem und synchronem Verhalten unterscheidet*; Diplomarbeit; Katholische Universität Eichstätt-Ingolstadt, 2006.
- [MBH+08] Mori, R.d., Bechet, F., Hakkani-Tur, D., McTear, M., Riccardi, G. und Tur, G.; *Spoken language understanding*; *IEEE Signal Processing Magazine* 25 (3), S. 50–58, 2008.

- [MBIS94] Miller, S., Bobrow, R., Ingria, R. und Schwartz, R.; *Hidden understanding models of natural language*; in: *Proceedings of the 32nd annual meeting on Association for Computational Linguistics*; S. 25–32, 1994.
- [MCCD13] Mikolov, T., Chen, K., Corrado, G. und Dean, J.; *Efficient Estimation of Word Representations in Vector Space*; CoRR abs/1301.3781, 2013.
- [MCLS08] Màrquez, L., Carreras, X., Litkowski, K.C. und Stevenson, S.; *Semantic Role Labeling: An Introduction to the Special Issue*; *Comput. Linguist.* 34 (2), S. 145–159, 2008.
- [MDY⁺15] Mesnil, G., Dauphin, Y., Yao, K., Bengio, Y., Deng, L., Hakkani-Tur, D., He, X., Heck, L., Tur, G., Yu, D. und Zweig, G.; *Using Recurrent Neural Networks for Slot Filling in Spoken Language Understanding*; *Trans. Audio, Speech and Lang. Proc.* 23 (3), S. 530–539, 2015.
- [MFP00] McCallum, A., Freitag, D. und Pereira, F.C.N.; *Maximum Entropy Markov Models for Information Extraction and Segmentation*; in: *Proceedings of the Seventeenth International Conference on Machine Learning*; ICML '00; S. 591–598; Morgan Kaufmann Publishers Inc, San Francisco and CA and USA, 2000.
- [MGJ⁺09] Mairesse, F., Gasic, M., Jurcicek, F., Keizer, S., Thomson, B., Yu, K. und Young, S.; *Spoken language understanding from unaligned data using discriminative classification models*; in: *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*; S. 4749–4752, 2009.
- [MGJ⁺10] Mairesse, F., Gašić, M., Jurčiček, F., Keizer, S., Thomson, B., Yu, K. und Young, S.; *Phrase-based statistical language generation using graphical models and active learning*; in: *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*; S. 1552–1561, 2010.
- [MHDB13] Mesnil, G., He, X., Deng, L. und Bengio, Y.; *Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding*; in: *INTERSPEECH*; S. 3771–3775, 2013.
- [Min75] Minsky, M.L.; *A framework for representing knowledge*; in: P.H. Winston und B. Horn, Hrsg., *The psychology of computer vision*; Bd. 67; S. 211–277; McGraw-Hill New York, 1975.
- [Moh09] Mohri, M.; *Weighted Automata Algorithms*; in: M. Droste, W. Kuich und H. Vogler, Hrsg., *Handbook of Weighted Automata*; Monographs in Theoretical Computer Science. An EATCS Series; S. 213–254; Springer Berlin Heidelberg, 2009.
- [Möl10] Möller, S.; *Quality Engineering: Qualität kommunikationstechnischer Systeme*; Springer-Verlag Berlin Heidelberg, 2010.

- [Mor07] Mori, R.d.; *Spoken language understanding: a survey*; in: *Automatic Speech Recognition & Understanding, 2007. ASRU. IEEE Workshop on*; S. 365–376, 2007.
- [Mor08] Morrissey, S.; *Data-driven machine translation for sign languages*; Dissertation; Dublin City University, 2008.
- [MPR08] Mohri, M., Pereira, F. und Riley, M.; *Speech Recognition with Weighted Finite-State Transducers*; in: J. Benesty, M. Sondhi und Y. Huang, Hrsg., *Springer Handbook of Speech Processing*; S. 559–584; Springer Berlin Heidelberg, 2008.
- [MSBI94] Miller, S., Schwartz, R., Bobrow, R. und Ingria, R.; *Statistical language processing using hidden understanding models*; in: *Proceedings of the workshop on Human Language Technology*; S. 278–282, 1994.
- [MSW16] *Microsoft Project Oxford - Web Language Model APIs*; https://www.projectoxford.ai/web_lm, 19.01.2016.
- [MY14] Mairesse, F. und Young, S.; *Stochastic language generation in dialogue using factored language models*; Computational linguistics , 2014.
- [Nöt00] Nöth, W.; *Handbuch der Semiotik*; Metzler, Stuttgart; 2. Aufl., 2000.
- [ON02] Och, F.J. und Ney, H.; *Discriminative Training and Maximum Entropy Models for Statistical Machine Translation*; in: *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*; S. 295–302; Association for Computational Linguistics, Stroudsburg, 2002.
- [OR00] Oh, A.H. und Rudnicky, A.I.; *Stochastic language generation for spoken dialogue systems*; in: *Proceedings of the 2000 ANLP/NAACL Workshop on Conversational systems*; S. 27–32, 2000.
- [PE05] Paiva, D.S. und Evans, R.; *Empirically-based Control of Natural Language Generation*; in: *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*; S. 58–65; Association for Computational Linguistics, Stroudsburg, 2005.
- [PERW97] Pietra, S.D., Epstein, M., Roukos, S. und Ward, T.; *Fertility Models for Statistical Natural Language Understanding*; in: Philip R. Cohen und Wolfgang Wahlster, Hrsg., *35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics*; S. 168–173; Morgan Kaufmann Publishers / ACL, 1997.
- [PK08] Pfister, B. und Kaufmann, T.; *Sprachverarbeitung: Grundlagen Und Methoden Der Sprachsynthese Und Spracherkennung (Springer-Lehrbuch)*; Springer; 1 Aufl., 2008.
- [PMMH08] Papaspyrou, C., von Meyenn, A., Matthaeh, M. und Herrmann, B.; *Grammatik der deutschen Gebärdensprache aus der Sicht gehörloser Fachleute*; Signum, Seedorf, 2008.

- [PMS+01] Pla, F., Molina, A., Sanchis, E., Segarra, E. und García, F.; *Language Understanding Using Two-Level Stochastic Models with POS and Semantic Units*; in: V. Matoušek, P. Mautner, R. Mouček und K. Taušer, Hrsg., *Text, Speech and Dialogue*; S. 403–409; Springer Berlin Heidelberg, 2001.
- [Pri90] Price, P.J.; *Evaluation of Spoken Language Systems: The ATIS Domain*; in: *Proceedings of the Workshop on Speech and Natural Language*; S. 91–95; Association for Computational Linguistics, Stroudsburg, 1990.
- [PRW97] Papineni, K., Roukos, S. und Ward, T.; *Feature-based language understanding EUROSPEECH 1997, Rhodes, Greece, September 22-25, 1997*; in: G. Kokkinakis, N. Fakotakis und E. Dermatas, Hrsg., *Fifth European Conference on Speech Communication and Technology, EUROSPEECH 1997, Rhodes, Greece, September 22-25, 1997*; ISCA, 1997.
- [RAL+06] Rajman, M., Ailomaa, M., Lisowska, A., Melichar, M. und Armstrong, S.; *Extending the wizard of oz methodology for language-enabled multimodal systems*; in: *Proceedings of LREC (Vol. 6)*, 2006.
- [Rat02] Ratnaparkhi, A.; *Trainable approaches to surface natural language generation and their application to conversational dialog systems*; *Computer Speech & Language* 16 (3), S. 435–455, 2002.
- [RBMD06] Raymond, C., Béchet, F., Mori, R.d. und Damnati, G.; *On the use of finite state transducers for semantic interpretation*; *Speech Communication* 48 (3–4), S. 288–304, 2006.
- [RD00] Reiter, E. und Dale, R.; *Building Natural Language Generation Systems*; Cambridge University Press, New York and NY and USA, 2000.
- [Rei91] Reisig, W.; *Petri nets and algebraic specifications*; *Theoretical Computer Science* 80 (1), S. 1–34, 1991.
- [Rei10] Reiter, E.; *Natural Language Generation.*; in: A. Clark, Hrsg., *The handbook of computational linguistics and natural language processing*; S. 574–598; Wiley-Blackwell, 2010.
- [Rit13] Ritter, D.; *Zielsetzung, Bestimmung und Konstruktion einer mikrolokalen Grammatik mit openFST am Beispiel einer Home Entertainment Anwendung*; Zulassungsarbeit; Katholische Universität Eichstätt-Ingolstadt; Eichstätt, 2013.
- [RM95] Ramshaw, L. und Marcus, M.; *Text Chunking Using Transformation-Based Learning*; in: D. Yarovsky und K. Church, Hrsg., *Proceedings of the Third Workshop on Very Large Corpora*; S. 82–94; Association for Computational Linguistics, Somerset and New Jersey, 1995.
- [Ros15] Rosenstock, Rachel; *Glossennotation "PETER" [E-Mail]*, 16.09.2015.

- [RR07] Raymond, C. und Riccardi, G.; *Generative and discriminative algorithms for spoken language understanding* Communication Association, Antwerp, Belgium, August 27-31, 2007; in: *INTERSPEECH 2007*; S. 1605–1608; ISCA, 2007.
- [RW14] Ritter, D. und Wirsching, G.; *Konstruktion einer mikrolokalen Grammatik mit OpenFST am Beispiel einer Home-Entertainment-Anwendung*; in: R. Hoffmann, Hrsg., *Elektronische Sprachsignalverarbeitung 2014*; Bd. 71 von *Studentexte zur Sprachkommunikation*; TUDpress, Dresden, 2014.
- [Sch06] Schnell, M.; *Prosodiegenerierung für die datenbasierte Sprachsynthese*; Bd. 38 von *Studentexte zur Sprachkommunikation*; TUDpress, Dresden, 2006.
- [Sen92] Seneff, S.; *Robust parsing for spoken language systems*; in: *Acoustics, Speech, and Signal Processing, 1992. ICASSP-92., 1992 IEEE International Conference on*; Bd. 1; S. 189–192, 1992.
- [Sha48] Shannon, C.E.; *A mathematical theory of communication*; Bell System Technical Journal, The 27 (3), S. 379–423 und 623–656, 1948.
- [Ski57] Skinner, B.F.; *Verbal behavior*; The Century psychology series; Prentice-Hall, Englewood Cliffs and NJ, 1957.
- [SLM06] Sandler, W. und Lillo-Martin, D.; *Sign language and linguistic universals*; Cambridge University Press, 2006.
- [SM12] Sutton, C. und McCallum, A.; *An Introduction to Conditional Random Fields*; Foundations and Trends in Machine Learning 4 (4), S. 267–373, 2012.
- [SRBN06] Servan, C., Raymond, C., Béchet, F. und Nocéra, P.; *Conceptual decoding from word lattices: application to the spoken dialogue corpus MEDIA*; in: *The Ninth International Conference on Spoken Language Processing (Interspeech 2006 - ICSLP)*; Pittsburgh and United States, 2006.
- [SSG+02] Segarra, E., Sanchis, E., Galiano, M., Garcia, F. und Hurtado, L.; *Extracting Semantic Information Through Automatic Learning Techniques*; International Journal of Pattern Recognition and Artificial Intelligence 16 (03), S. 301–307, 2002.
- [SSN12] Stein, D., Schmidt, C. und Ney, H.; *Analysis, Preparation, and Optimization of Statistical Sign Language Machine Translation*; Machine Translation 26 (4), S. 325–357, 2012.
- [ST95] Schukat-Talamazzini, E.G.; *Automatische Spracherkennung: Grundlagen, statistische Modelle und effiziente Algorithmen*; Artificial intelligence Künstliche Intelligenz; F. Vieweg, Braunschweig, 1995.
- [Sta16] *EMNLP 2016 - Shared Task: Machine Translation*; <http://www.statmt.org/wmt16/translation-task.html>, 2016; Stand: 20.03.2016.

- [Ste06] Stein, D.; *Morpho-syntax Based Statistical Methods for Sign Language Translation*; Dissertation; Rheinisch-Westfälische Technische Hochschule Aachen, 2006.
- [Ste12] Stein, D.; *Soft features for statistical machine translation of spoken and signed languages*; Dissertation; Aachen, Techn. Hochsch; Aachen, 2012.
- [Sto93] Storey, V.C.; *Understanding semantic relationships*; The VLDB Journal 2 (4), S. 455–488, 1993.
- [Stö14] Stöhr, M.; *Wizard-of-Oz-Methode - Sprachsteuerung einer Musikbibliothek.*; Bachelorarbeit; BTU Cottbus - Senftenberg, 2014.
- [Str13] Streitböcker, W.; *Grundbegriffe für Journalistenausbildung: Theorie, Praxis und Techniken als berufliche Techniken*; Research; Springer, 2013.
- [SW49] Shannon, C.E. und Weaver, W.; *The mathematical theory of communication*; University of Illinois Press, 1949.
- [Tab06] Tabeling, P.; *Softwaresysteme und ihre Modellierung: Grundlagen, Methoden und Techniken*; EXamen.press; Springer, Berlin, 2006.
- [TKP+01] Theune, M., Klabbbers, E., Pijper, J.R.d., Kraemer, E. und Odijk, J.; *From Data to Speech: A General Approach*; Nat. Lang. Eng. 7 (1), S. 47–86, 2001.
- [TSV+10] Tur, G., Stolcke, A., Voss, L., Peters, S., Hakkani-Tür, D., Dowding, J., Favre, B., Fernández, R., Frampton, M., Frandsen, M. et al.; *The CALO meeting assistant system*; Audio, Speech, and Language Processing, IEEE Transactions on 18 (6), S. 1601–1611, 2010.
- [TWH13] Tur, G., Wang, Y.Y. und Hakkani-Tur, D.; *Understanding Spoken Language*; in: A.B. Tucker, T.F. Gonzalez, H. Topi und J.L. Diaz-Herrera, Hrsg., *Computing handbook*; Chapman and Hall/CRC Press, 2013.
- [UAS] *Softwareprojekt UASR: Unified Approach to Speech Synthesis and Recognition*; <https://github.com/matthias-wolff/UASR>; Stand: 20.03.2016.
- [UCU] *Universal Cognitive User Interface (2015 – 2018)*; <http://www.mtidw.de/ueberblick-bekanntmachungen/ALS/ucui>; Stand: 06.06.2016.
- [Uhl12] Uhlig, A.C.; *Ethnographie der Gehörlosen: Kultur - Kommunikation - Gemeinschaft*; Kultur und soziale Praxis; Transcript, Bielefeld, 2012.
- [VRG15] Vukotic, V., Raymond, C. und Gravier, G.; *Is it time to switch to Word Embedding and Recurrent Neural Networks for Spoken Language Understanding?*; in: *INTERSPEECH*; Dresden and Germany, 2015.
- [War91] Ward, W.; *Understanding spontaneous speech: the Phoenix system*; Acoustics, Speech, and Signal Processing, IEEE International Conference on 0, S. 365–367, 1991.

- [WBB+10] Wanner, L., Bohnet, B., Bouayad-Agha, N., Lareau, F.c. und Nicklaß, D.; *MARQUIS: Generation of user-tailored multilingual air quality bulletins*; Applied Artificial Intelligence 24 (10), S. 914–952, 2010.
- [WDA05] Wang, Y.Y., Deng, L. und Acero, A.; *Spoken language understanding*; Signal Processing Magazine, IEEE 22 (5), S. 16–31, 2005.
- [WGK+15] Wen, T.H., Gašić, M., Kim, D., Mrkšić, N., Su, P.H., Vandyke, D. und Young, S.; *Stochastic Language Generation in Dialogue using Recurrent Neural Networks with Convolutional Sentence Reranking*; in: *Sigdial 2015*, 2015.
- [WHK+12] Wirsching, G., Huber, M., Kölbl, C., Lorenz, R. und Römer, R.; *Semantic Dialogue Modeling*; in: A. Esposito, A.M. Esposito, A. Vinciarelli, R. Hoffmann und V.C. Müller, Hrsg., *Cognitive behavioural systems*; Bd. 7403 von *Lecture Notes in Computer Science*; S. 104–113; Springer, Berlin and Heidelberg, 2012.
- [WI94] Ward, W. und Issar, S.; *Recent Improvements in the CMU Spoken Language Understanding System*; in: *Proceedings of the Workshop on Human Language Technology*; S. 213–216; Association for Computational Linguistics, Stroudsburg, 1994.
- [Wir12] Wirsching, G.; *Nichtsequentialität in der Sprachverarbeitung mit FST*; in: M. Wolff, Hrsg., *Elektronische Sprachsignalverarbeitung 2012*; Bd. 64 von *Studenten zur Sprachkommunikation*; S. 26–33; TUDpress, Dresden, 2012.
- [WKTS91] Wang, F., Kyriakopoulos, K., Tsolkas, A. und Saridis, G.; *A Petri-net coordination model for an intelligent mobile robot*; Systems, Man and Cybernetics, IEEE Transactions on 21 (4), S. 777–789, 1991.
- [WL13] Wirsching, G. und Lorenz, R.; *Towards meaning-oriented language modeling*; in: *Cognitive Infocommunications (CogInfoCom), 2013 IEEE 4th International Conference*; S. 369–374, 2013.
- [WM07] Wong, Y.W. und Mooney, R.; *Generation by Inverting a Semantic Parser that Uses Statistical Machine Translation*; in: *Human Language Technologies*; S. 172–179; Association for Computational Linguistics, Rochester and New York, 2007.
- [Woo70] Woods, W.A.; *Transition network grammars for natural language analysis*; Communications of the ACM 13 (10), S. 591–606, 1970.
- [WRH12] Wolff, M., Römer, R. und Hoffmann, R.; *Hierarchische kognitive dynamische Systeme zur Sprach- und Signalverarbeitung*; in: M. Wolff, Hrsg., *Elektronische Sprachsignalverarbeitung 2012*; Bd. 64 von *Studenten zur Sprachkommunikation*; S. 96–103; TUDpress, Dresden, 2012.

- [WS89] Wang, F. und Saridis, G.; *A model for coordination of intelligent machines using Petri nets*; in: J.Y.S. Luh, Hrsg., *Intelligent Control*; S. 28–33; IEEE Computer Society Press, 1989.
- [WTV⁺10] Wang, K., Thrasher, C., Viegas, E., Li, X. und Hsu, B.j.; *An Overview of Microsoft Web N-gram Corpus and Applications*; in: *Proceedings of the NAACL HLT 2010 Demonstration Session*; S. 45–48; Association for Computational Linguistics, Stroudsburg, 2010.
- [WW14] Wirsching, G. und Wolff, M.; *Semantische Dekodierung von Sprachsignalen am Beispiel einer Mikrofonfeldsteuerung*; in: R. Hoffmann, Hrsg., *Elektronische Sprachsignalverarbeitung 2014*; Bd. 71 von *Studentexte zur Sprachkommunikation*; S. 104–109; TUDpress, Dresden, 2014.
- [XAN03] Xu, Y., Araki, M. und Niimi, Y.; *A multilingual-supporting dialog system across multiple domains*; *Acoustical Science and Technology* 24 (6), S. 349–357, 2003.
- [YGTW13] Young, S., Gasic, M., Thomson, B. und Williams, J.; *POMDP-Based Statistical Spoken Dialog Systems: A Review*; *Proceedings of the IEEE* 101 (5), S. 1160–1179, 2013.
- [You02] Young, S.; *The statistical approach to the design of spoken dialogue systems: Technical Report CUED/F-INFENG/TR.433, Cambridge University Engineering Department*, 2002.
- [You10] Young, S.; *Cognitive User Interfaces*; *Signal Processing Magazine, IEEE* 27 (3), S. 128–140, 2010.
- [YZH⁺13] Yao, K., Zweig, G., Hwang, M.Y., Shi, Y. und Yu, D.; *Recurrent neural networks for language understanding*; in: *INTERSPEECH*; S. 2524–2528, 2013.

