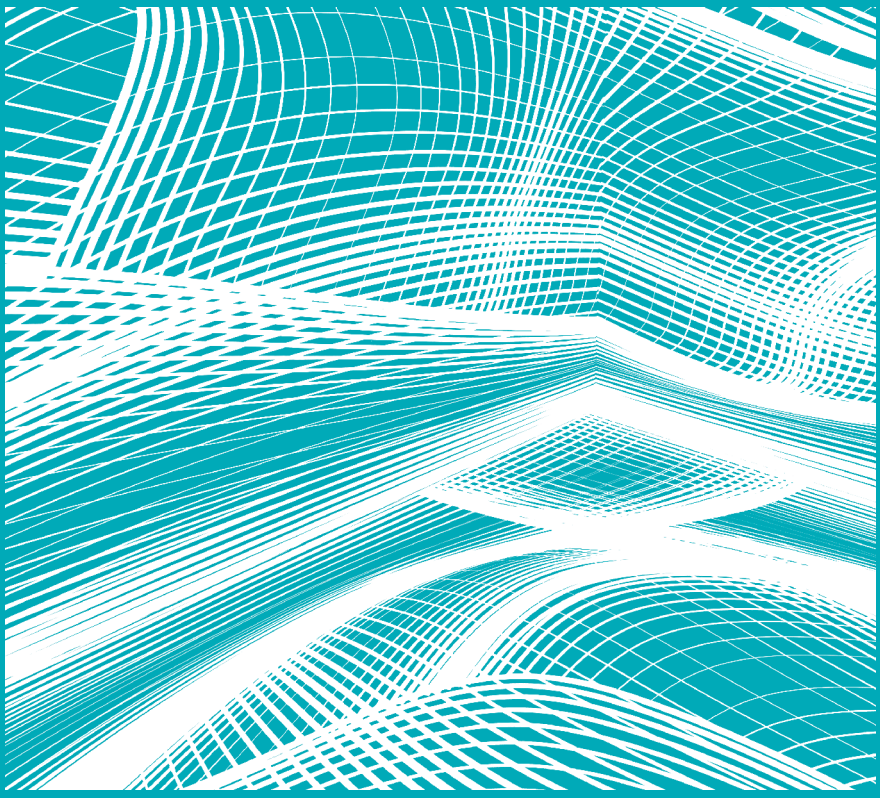
CLAES NEUEFEIND

Muster und Bedeutung

Bedeutungskonstitution als kontextuelle Aktivierung im Vektorraum



MAP

Herausgegeben von
Modern Academic Publishing (MAP)
2019

MAP (Modern Academic Publishing) ist eine Initiative an der Universität zu Köln, die auf dem Feld des elektronischen Publizierens zum digitalen Wandel in den Geisteswissenschaften beiträgt. MAP ist angesiedelt am Lehrstuhl für die Geschichte der Frühen Neuzeit von Prof. Dr. Gudrun Gersmann.

Die MAP-Partner Universität zu Köln (UzK) und Ludwig-Maximilians-Universität München (LMU) fördern die Open-Access-Publikation von Dissertationen forschungsstarker junger Geisteswissenschaftler beider Universitäten und verbinden dadurch wissenschaftliche Nachwuchsförderung mit dem Transfer in eine neue digitale Publikationskultur.

www.humanities-map.net

Claes Neuefeind

Muster und Bedeutung

Bedeutungskonstitution als kontextuelle Aktivierung im Vektorraum

MAP©
Modern Academic Publishing

Herausgegeben von
Modern Academic Publishing
Universität zu Köln
Albertus-Magnus-Platz
50923 Köln

Gefördert von der Universität zu Köln

Text © Claes Neuefeind 2019
Diese Arbeit ist veröffentlicht unter Creative Commons Licence BY-SA 4.0. Eine Erläuterung zu dieser Lizenz findet sich unter http://creativecommons.org/licenses/by/4.0/. Diese Lizenz erlaubt die Weitergabe aus der Publikation unter gleichen Bedingungen für privaten oder kommerziellen Gebrauch bei ausreichender Namensnennung des Autors. Grafiken, Tabellen und Abbildungen unterliegen ggf. eigenen Lizenzen, die jeweils angegeben und gesondert zu berücksichtigen sind.

Erstveröffentlichung 2019
Zugleich Dissertation der Universität zu Köln 2017
Umschlagbild: Armand Khoury, Ohne Titel, Foto, https://unsplash.com/photos/4cBVro7SHLs, CC BY.

Bibliografische Information der Deutschen Nationalbibliothek
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über http://dnb.dnb.de abrufbar.

ISBN (Hardcover): 978-3-946198-40-6
ISBN (EPUB): 978-3-946198-41-3
ISBN (Kindle): 978-3-946198-42-0
ISBN (PDF): 978-3-946198-43-7
DOI: https://doi.org/10.16994/bam

Herstellung & technische Infrastruktur:
Ubiquity Press Ltd, 6 Osborn Street, Unit 2N, London E1 6TD, United Kingdom

Open Access-Version dieser Publikation verfügbar unter:
https://doi.org/10.16994/bam
oder Einlesen des folgenden QR-Codes mit einem mobilen Gerät:

Inhalt

Danksagung		IX
English Summary		XI
1.	**Einleitung**	**1**
1.1	Gegenstand und Zielsetzung	3
1.2	Aufbau der Arbeit	5
2.	**Das Bedeutungspotential sprachlicher Einheiten**	**9**
2.1	Die Variabilität sprachlicher Bedeutung	9
2.1.1	Ambiguität	10
2.1.2	Bedeutungsvariation	13
2.2	Zusammenfassung	14
3.	**Bedeutungspotential und Bedeutungskonstitution**	**17**
3.1	Kognitive Linguistik	17
3.1.1	Holistischer Ansatz	19
3.1.2	Sprache als semantisches Wissen	22
3.2	Kognitive Semantik	23
3.2.1	Bedeutung als Potential	24
3.2.2	Bedeutung als Prozess	28
3.2.3	Implikationen für die Modellierung	32
3.3	Zusammenfassung	35
4.	**Das Word Space Model**	**39**
4.1	Grundkonzeption des Modells	40
4.1.1	Der Wortraum	40
4.1.2	Wörter als Vektoren	42
4.1.3	Kontextvektoren und Kookkurrenzen zweiter Ordnung	47
4.1.4	Zusammenfassung	49
4.2	Theoretische Grundlagen des Modells	50
4.2.1	Der Word Space als semantischer Raum	51
4.2.2	Die distributionelle Hypothese	52
4.2.3	Diskussion	54
4.3	Zusammenfassung	56
5.	**Bedeutungskonstitution im Vektorraum**	**59**
5.1	Repräsentation von Input und Output	59
5.1.1	Bedeutungspotential im Vektorraum	60
5.1.2	Input und Output als Vektoren	61

5.2 Bedeutungskonstitution als Transformation von Vektoren 63
5.2.1 Transformation durch den Kontext 64
5.2.2 Gewichtung der Kontexte 68
5.2.3 Mehrdeutigkeit im Vektorraum 71
5.3 Diskussion 73

6. Softwaretechnologische Umsetzung 77
6.1 Das Text Engineering Software Laboratory (Tesla) 78
6.1.1 Experimente im virtuellen Labor 79
6.1.2 Arbeiten im virtuellen Labor 80
6.1.3 Das Tesla Role System 84
6.2 Verfahrensschritte und Komponenten 86
6.2.1 Korpora 87
6.2.2 Vorverarbeitung 88
6.2.3 Kookkurrenzvektoren 89
6.2.4 Normalisierung 90
6.2.5 Gewichtung 91
6.2.6 Token-Vektoren 92
6.2.7 Clusteranalyse 93
6.2.8 Visualisierung 95
6.2.9 Beispielwörter für die Experimente 97
6.3 Zusammenfassung 101

7. Experimente zur Bedeutungskonstitution 105
7.1 Repräsentation der Eingabeinformation 105
7.1.1 Aufbau des Experiments 106
7.1.2 Parametrisierung 108
7.1.3 Referenzräume 110
7.2 Bedeutungskonstitution in Einzelkontexten 114
7.2.1 Aufbau des Experiments 115
7.2.2 Parametrisierung 117
7.2.3 Beispielanalysen 118
7.3 Semantische Profile 126
7.3.1 Aufbau des Experiments 127
7.3.2 Parametrisierung 127
7.3.3 Beispielanalysen 130
7.4 Zusammenfassung 135

8. Fazit: Muster und Bedeutung 139

A. Komponenten 143
A.1 Reader 143
A.1.1 LCC Reader 143

A.1.2 SdeWaC Reader 143
A.2 Vorverarbeitung 144
A.2.1 Simple Tokenizer 144
A.2.2 Tree Tagger Wrapper 144
A.2.3 Snowball Stemmer Wrapper 145
A.3 Vektorerstellung 145
A.3.1 Sentence Based Vector Generator 145
A.3.2 Punctuation Filter 146
A.3.3 Frequency Range Filter 146
A.3.4 POSFilter 147
A.3.5 Wordlist Filter 148
A.4 Normalisierung und Gewichtung 148
A.4.1 VectorNormalization 148
A.4.2 VectorWeighting 148
A.5 Repräsentation von Einzelvorkommen 149
A.5.1 CollocationVectors 149
A.5.2 Context Vectors 150
A.5.3 Sentence Vectors 151
A.6 Clustering 152
A.6.1 Distanzbasierte Verfahren 152
A.6.2 Dichtebasierte Verfahren 154
A.6.3 ClusterFilter 155
A.7 Visualisierung 155

B. Experimente 157
B.1 Kookkurrenzvektoren und Referenzräume 158
B.2 Bedeutungskonstitution in Einzelkontexten 160
B.3 Semantische Profile 161

C. Assoziationsmaße 163
C.1 Pointwise Mutual Information 163
C.2 Log-Likelihood-Ratio 164

Abbildungsverzeichnis 167

Literaturverzeichnis 169

Danksagung

Der vorliegende Text ist eine leicht überarbeitete Fassung meiner Dissertation, die im Februar 2017 von der Philosophischen Fakultät der Universität zu Köln angenommen wurde. Mein Dank gilt zuerst meinem Doktorvater Prof. Dr. Jürgen Rolshoven für die zahlreichen anregenden Gespräche, für seine Ideen und seinen Rat, für die uneingeschränkte Unterstützung und das große Vertrauen, das er mir stets entgegengebracht hat – ohne ihn wäre diese Arbeit nicht möglich gewesen. Ebenfalls großer Dank gebührt meinem Zweitgutachter Prof. Dr. Dr. h.c. Andreas Speer, nicht zuletzt auch stellvertretend für die a.r.t.e.s. Graduate School for the Humanities, die mir ein ideales Umfeld war, um meine Ideen zu diskutieren und sie auch über das eigene Fach hinaus zu reflektieren und zu schärfen. Hierfür danke ich den Mentorinnen Prof. Dr. Chris Bongartz und Prof. Dr. Claudia Riehl sowie all meinen ehemaligen Kommilitonen bei a.r.t.e.s., insbesondere Dr. Reinhard Messerschmidt, der mir in der Schlussphase der Dissertation ein unverzichtbarer Gesprächspartner und Motivator war. Den Herausgebern Prof. Dr. Gudrun Gersmann und Prof. Dr. Hubertus Kohle, der Universität zu Köln sowie der LMU München möchte ich für die Möglichkeit danken, meine Dissertation über Modern Academic Publishing in diesem innovativen Format zu publizieren. Hierbei möchte ich insbesondere Dr. Claudie Paye, Christine Schmitt und Ann Catrin Bolton danken für ihre professionelle und tatkräftige Unterstützung während des gesamten Publikationsprozesses. Ganz herzlich danken möchte ich zudem auch meinen ehemaligen Kollegen in der Sprachlichen Informationsverarbeitung: Allen voran Dr. Stephan Schwiebert, der mir eine unverzichtbare Hilfe bei der Umsetzung dieser Arbeit war, indem er mir auch aus dem fernen Australien noch bis zuletzt viele wertvolle Hinweise und inhaltliche Anmerkungen gab. Auch danke ich Francisco Mondaca für die engagierte Mithilfe bei der finalen Fassung, Mona Weinle für ihre sorgfältigen Korrekturen und Anne Pietsch für die Hilfe bei der Neuformatierung für diese Publikation; mein Dank gilt außerdem Mihail Atanassov und Fabian Steeg für die produktive und angenehme Zusammenarbeit in den gemeinsamen Projekten, und gleichermaßen auch Alena Geduldig, Dr. Jürgen Hermes, Börge Kiss, David Neugebauer, David Rival, Peter Seipel sowie allen ehemaligen Hilfskräften, die alle zusammen die Entstehung dieser Arbeit mit unzähligen Pausengesprächen sowie mit viel Geduld, Verständnis und aufmunternden Worten begleitet haben. Ein ganz besonderer Dank gilt meinen Freunden, die mir stets Abstand, Ausgleich und neue Motivation zu geben vermochten. Und von ganzem Herzen danke ich meiner Familie: Meinen Eltern sowie meinen Schwestern für ihre geduldige Unterstützung und ihren anhaltenden Glauben an die Fertigstellung dieser Arbeit.

Mein größter Dank gilt jedoch Anja, meiner großen Liebe, und unseren wundervollen Töchtern Leni, Merle und Lisbeth. Sie sind das Fundament, ohne das ich die Ausdauer und Energie für diese Arbeit nicht hätte aufbringen können.

Köln, im März 2018 Claes Neuefeind

English Summary

From patterns to meaning
Meaning Constitution as contextual activation in vector space

The subject of this thesis is a computational linguistic model of Meaning Constitution in linguistic units. Taking the phenomenon of variability of linguistic meaning as its starting point, Meaning Constitution is described as an information-processing step, which is then implemented and empirically tested in a series of linguistic experiments. In this thesis, Meaning Constitution is understood as a dynamic process in which the meaning of linguistic units only becomes concrete within local contexts in relation to their general meaning potential. This dynamic concept of meaning is based on a central assumption of Cognitive Semantics, according to which meanings do not exist independently of the context. The motivation for the implementation of a computational linguistic model of its own is the fact that the conception of meaning in Cognitive Semantics itself does not involve such an operationalisation – which, strictly speaking, means that it must be regarded as not falsifiable.

The modelling is carried out against the background of the Distributional Hypothesis according to Zellig Harris. By algorithmically extracting linguistic patterns and their relations in large text corpora, a representation of the meaning potential is made by means of vectors in word space. Based on these, the Meaning Constitution is modelled as an information-processing step, in the course of which a local adaptation of the initial representations takes place. The notion of pattern plays a central role here: Interpreted as patterns of use, it forms the basis both for the representation of the meaning potential and for the actual modelling of the process of Meaning Constitution.

By including the process of Meaning Constitution, an interpretation of the word space is made within this thesis, which deviates from the common structuralistic interpretation. Instead, the patterns of use encoded by the word vectors are transferred into the theoretical framework of Cognitive Semantics. Although the patterns of use are by themselves not suitable for explaining the dynamic conception of meaning of Cognitive Semantics, the patterns of use do also play a decisive role from a cognitive perspective, as they form the starting point for the process of Meaning Constitution. The patterns of use can thus be understood as a building block of semantic memory, on the basis of which the concrete meanings are formed locally. In the model proposed here, the patterns of use are therefore the decisive information carrier and supplier. In other words: when there is no pattern, there is no meaning.

The methodological principle guiding this thesis is an empirical-experimental approach to linguistic problems. The requirements to be considered for scientific experiments – control, reproducibility and variation – are taken into account by

means of the software-technological implementation within the Text Engineering Software Laboratory (Tesla, see http://tesla.spinfo.uni-koeln.de). Tesla is a linguistic component system developed in the Linguistic Information Processing department at the University of Cologne. In analogy to a scientific laboratory, Tesla offers the possibility to segment and annotate textual data within experimental arrangements and to apply linguistically motivated computational methods. Tesla thus takes on the function of a virtual laboratory, in which the model is tested in a series of virtual experiments in order to draw conclusions about the explicative value of the underlying dynamic concept of meaning.

The main objective of the computational linguistic experiments is to show, by means of exemplary analyses of selected words, how the dynamic concept of meaning of Cognitive Linguistics can be modelled as a contextual activation in vector space via the process of Meaning Constitution. By examining ambiguous linguistic units, it is shown that the constitution of meaning can be understood as a process of the development of complex linguistic patterns. Beyond the experimental testing of the computational model, the connection between pattern formation and meaning constitution becomes the object of the investigation. The central assumption is that meaning can be modelled by a transformation of the extracted patterns of use. This also raises the question of the conditions and possibilities of a purely data-driven approach to the problem of determining meaning; this applies in particular to the question of the suitability of a purely distributional methodology for modelling a dynamic concept of meaning in the sense of the theoretical assumptions of Cognitive Linguistics.

In this thesis, knowledge about linguistic systems is not seen as a prerequisite, but rather as the consequence and result of the systematic analysis. In this context, information-processing systems are a central component of linguistic theory development, insofar as their use makes it possible to make contexts and conditions of use accessible for systematic analysis, independent of the implicit prior knowledge of human agents. Being located between fundamental linguistic research and computational linguistic application, this thesis illustrates the role of computational linguistics in cognitive science, particularly with regard to the modelling of a cognitively motivated theory of meaning: by enabling the simulation of cognitive processes and by providing tools for the empirical-experimental testing of the associated models, computational linguistics itself plays a central role in the formation of linguistic theory. With the formulation of concrete linguistic experiments and by providing the corresponding procedures and results by means of Tesla, the computational linguistic modelling of Meaning Constitution in the course of this thesis is meant to be a contribution to a better understanding of the semantic dynamics of language.

1. Einleitung

Dies ist eine computerlinguistische Arbeit. Auch wenn die Computerlinguistik (CL) als Disziplin auf eine mittlerweile über 60-jährige Tradition zurückblickt, ist sie in ihrer Ausrichtung bis heute kein einheitlicher Bereich. Zum einen speist sie sich, wie bereits der Name verrät, aus verschiedenen Disziplinen, zum anderen gibt es auch eine grundsätzliche Unterscheidung in Bezug auf die inhaltliche und methodische Ausrichtung. Dies schlägt sich unter anderem darin nieder, dass die CL zwar oftmals an Informatik-Lehrstühlen angesiedelt ist, in einigen Fällen – so auch in Köln – jedoch mit einer stärker geisteswissenschaftlichen Ausrichtung in der Linguistik verortet ist. Aus diesen Gründen erscheint es angebracht, hier zunächst eine Perspektivierung vorzunehmen. In Bezug auf die Aufgabe der CL lassen sich im Wesentlichen zwei Perspektiven unterscheiden: Auf der einen Seite ist die CL eine angewandte Informatik, die eine Modellierung konkreter Anwendungsfälle zum Gegenstand hat, etwa die Informationssuche, Maschinelle Übersetzung, etc.; auf der anderen Seite ist die CL als Teilbereich der Kognitionswissenschaften anzusehen.

In der ersten Lesart ist im Wesentlichen der Bereich der Maschinellen Sprachverarbeitung gemeint (Natural Language Processing, NLP), welche als ein Teilbereich der Künstlichen Intelligenz (KI) angesehen werden kann, mit der sie von Beginn an eng verzahnt war.[1] Die KI zielt auf den Entwurf und die Umsetzung intelligenter Systeme; in dieser primär anwendungsorientierten Ausrichtung erfolgt der Systementwurf in der Regel stärker ergebnisorientiert. Informationsverarbeitende Prozesse werden hier häufig vom angestrebten Resultat aus gedacht, so dass es in vielen Fällen gute Gründe gibt, pragmatische Entscheidungen zu treffen, etwa bereits bestehende Ressourcen zu nutzen oder verfügbare Ansätze zu integrieren und bedarfsgerecht anzupassen. Damit verbunden sind oftmals vorgelagerte Theorieentscheidungen, welche den Systementwurf maßgeblich beeinflussen – oftmals ohne dass dies expliziert wird.

In der zweiten Lesart ist die CL ein methodisches Instrument der Kognitionswissenschaften, speziell der Teildisziplin der Kognitiven Linguistik, die sich mit Modellen des Sprachverstehens, der Sprachproduktion und des Spracherwerbs beschäftigt. In Bezug auf die Kognitive Linguistik übernimmt die CL nach Rickheit u.a. (2010, 193) eine »methodische Funktion, die durch den Computer als Werkzeug bestimmt ist«. In dieser Perspektive hat die CL die Simulation von Modellen sprachverarbeitender kognitiver Prozesse zum Ziel sowie die experimentelle

1 Siehe dazu z.B. Russell/Norvig (2012, 36): »Die moderne Linguistik und die KI wurden also etwa gleichzeitig ›geboren‹ und wuchsen zusammen auf, mit einer Schnittmenge in einem hybriden Gebiet, der sogenannten Computerlinguistik oder natürlichen Sprachverarbeitung«. Die im Zitat genannte »natürliche Sprachverarbeitung« ist dabei eine eher unübliche Übersetzung des Terminus Natural Language Processing (NLP), es handelt sich somit um nichts anderes als die Maschinelle Sprachverarbeitung.

Evaluierung dieser Modelle – wobei Modelle »nicht nur aus der Menge der von ihnen beschriebenen Entitäten, sondern auch aus den Prozessen, die für die Beschreibung der Abläufe im Modell zuständig sind« (Rickheit u.a. 2010, 196) bestehen. Insbesondere für neuere Ansätze der Kognitiven Linguistik, die eine stärker empirisch geprägte Ausrichtung verfolgen, ist eine solche methodische Ergänzung von zentraler Bedeutung, da diesen oftmals ein entsprechendes methodisches Fundament fehlt.[2] So weisen etwa Rickheit u.a. (2010) explizit auf die Notwendigkeit empirischer Forschung in der Kognitiven Linguistik hin. Hierbei beziehen sie sich unter anderem auf Evans/Green (2006, 781f.), die – hier wiedergegeben mit den Worten von Rickheit u.a. – »[…] beanstanden, dass viele Theorien der Kognitiven Linguistik nicht empirisch überprüfbar und falsifizierbar sind, was wissenschaftstheoretisch als Voraussetzung für eine Theorie betrachtet wird. Andernfalls handelt es sich um eine bloße Ideologie oder Spekulation« (Rickheit u.a. 2010, 14).

Eine differenzierte Auseinandersetzung mit der Rolle der Modellierung für die linguistische Theoriebildung findet sich unter anderem auch bei Burghard Rieger, der sich in einer Reihe von Arbeiten dem Problem der Modellierung eines kognitiv motivierten, als hochgradig dynamisch anzusehenden Bedeutungsbegriffs widmet (siehe unter anderem Rieger 1977; 1980; 1985; 1989). In der Einleitung zum Sammelband »Dynamik in der Bedeutungskonstitution«[3] fasst Rieger das Verhältnis von Theorie, Modell und Experiment wie folgt zusammen:

> Dabei läßt sich unterscheiden zwischen den *Theorien*, die allgemeine und umfassende Zusammenhänge formulieren, den daraus entwickelten *Modellen*, die kleinere und überschaubare Ausschnitte dieser Zusammenhänge abbilden, und der experimentellen *Erprobung* dieser Modelle, welche als Überprüfung und Vergleich von Daten, Test von Hypothesen, Analyse von Strukturen, Simulation von Prozessen, [sic!] etc. erst Rückschlüsse auf den explikativen Wert der Theorie zu ziehen erlaubt. (Rieger 1985, 1; Hervorhebungen gemäß Original)

Die CL ist in dieser Sicht ein methodischer Ansatz zur Sprachtheorie, der in erster Linie darin besteht, Werkzeuge bereitzustellen, die eine Modellierung von sprachverarbeitenden Prozessen ermöglichen, gleichsam als »virtuelles Labor, in dem virtuelle Experimente durchgeführt werden« (Rickheit u.a. 2010, 196). Ebendiese Vorstellung eines virtuellen Labors ist auch das zentrale Konzept des Text Engineering Software Laboratory (Tesla), dem in dieser Arbeit eine wesentliche Rolle zukommt. Tesla ist ein linguistisches Komponentensystem, das in der Sprachlichen Informationsverarbeitung an der Universität zu Köln entwickelt wurde.[4]

2 Dies liegt u.a. auch daran, dass es sich bei der empirischen Ausrichtung um einen relativ jungen Ansatz innerhalb der Kognitiven Linguistik handelt, der sich in stetiger Weiterentwicklung befindet.

3 Der Sammelband enthält die Beiträge der eingeladenen Teilnehmer der Semantik-Sektion des Deutschen Germanistentags 1982 in Aachen (Rahmenthema: »Bedeutungskonstitution. Beschreibung, Analyse und Simulation von Sprachproduktions- und Verstehensprozessen«).

4 Siehe http://tesla.spinfo.uni-koeln.de (Zugriff vom 04.09.2017); Schwiebert (2012); Hermes (2012).

Analog zu einem naturwissenschaftlichen Labor bietet Tesla Möglichkeiten, textuelle Daten innerhalb von experimentellen Anordnungen zu segmentieren, auszuzeichnen und computerlinguistisch motivierte, unter anderem etwa musterbildende Verfahren darauf anzuwenden. Die Experimente werden vollständig und automatisch in einem virtuellen ›Laborheft‹ dokumentiert; dabei wird zusammen mit den Ergebnissen der Experimente auch der gesamte Versuchsaufbau gespeichert, bestehend aus der Auswahl an Ausgangsdaten und den für die Verarbeitung eingesetzten Software-Komponenten, einschließlich ihrer Versionsnummer, Konfiguration und der jeweiligen experimentellen Anordnung. Durch diese Art der Dokumentation sind die Ergebnisse der Experimente jederzeit reproduzierbar, etwa um experimentelle Ausgänge zu überprüfen, die Verfahren auf eine andere Datenbasis anzuwenden oder um die Parameter in den eingesetzten Komponenten zu modifizieren. Dadurch können in Tesla – ganz im Sinne von Riegers Unterscheidung von Theorie, Modell und Experiment – die den Experimenten zugrunde gelegten Modelle erprobt, Hypothesen getestet und Prozesse simuliert werden, um daraus Rückschlüsse auf die theoretische Konzeption zu ziehen.

1.1 Gegenstand und Zielsetzung

Im Mittelpunkt dieser Arbeit steht eine computerlinguistische Modellierung der Bedeutungskonstitution in sprachlichen Einheiten. Bedeutungskonstitution wird in dieser Arbeit als dynamischer Prozess verstanden, bei dem sich die Bedeutung sprachlicher Einheiten erst innerhalb lokaler Kontexte in Relation zu deren allgemeinem Bedeutungspotential konkretisiert. Diese Konzeption eines dynamischen Bedeutungsbegriffs nimmt Überlegungen aus neueren Ansätzen der Kognitiven Semantik auf und stützt sich dabei insbesondere auf den *dynamic construal approach* von Alan Cruse (siehe Croft/Cruse 2004; Cruse 2011).[5]

Zentrales Motiv für die Umsetzung eines eigenen computerlinguistischen Modells ist die Tatsache, dass die Konzeption von Croft/Cruse (2004) selbst keine entsprechende Operationalisierung der Bedeutungskonstitution beinhaltet,[6] weshalb sie streng genommen als nicht falsifizierbar anzusehen ist und somit gemäß der oben angestellten Vorüberlegungen als »bloße Ideologie oder Spekulation« (Rickheit u.a. 2010, 14) angesehen werden könnte. Die Modellierung erfolgt in dieser Arbeit unter Rückgriff auf das Word Space Model (WSM) nach Schütze

5 Eine ähnliche Konzeption findet sich u.a. auch in den Arbeiten von Burghard Rieger (vgl. Rieger 1985; 1989). In gewisser Weise sind Riegers Arbeiten demnach als eine frühe Ausformulierung der Positionen einer empirisch ausgerichteten Kognitiven Linguistik anzusehen. Unterschiede bestehen jedoch u.a. in der Terminologie: wo Rieger (1985) von semantischen Dispositionen spricht, einem Begriff aus der Verhaltenspsychologie mit einer deutlichen sozio-psychologischen Konnotation, wird in dieser Arbeit der etwas neutralere Begriff des Bedeutungspotentials verwendet.

6 Anders als z.B. Rieger, dessen Ansatz auf einer Kombination aus statistischer Korrelationsanalyse und Konzepten der *Fuzzy Sets* (Zadeh 1965) basiert – dies ist jedoch nicht Gegenstand dieser Arbeit.

(1992; 1993).[7] Im WSM erfolgt zunächst die Repräsentation des Bedeutungspotentials über die algorithmische Erfassung von sprachlichen Mustern (konkret: von Verwendungsmustern) und ihren Relationen in großen Textkorpora. Auf dieser Grundlage lässt sich in einem weiteren Schritt der Prozess der Bedeutungskonstitution in Form gängiger Vektoroperationen realisieren. Der Begriff des Musters nimmt damit eine zentrale Rolle in dieser Arbeit ein: Im Sinne von Verwendungsmustern bildet er die Grundlage sowohl für die Repräsentation im WSM als auch für den eigentlichen Prozess, im Zuge dessen die Bedeutungskonstitution durch eine lokale Anpassung der (Verwendungs-)Muster erfolgt.

Methodischer Leitgedanke des Vorhabens ist eine empirisch-experimentelle Herangehensweise an sprachwissenschaftliche Problemstellungen. Die dabei nach Rickheit u.a. (2010, 196) zu beachtenden Anforderungen an wissenschaftliche Experimente – Kontrolle, Wiederholbarkeit und Variation – werden durch die softwaretechnologische Umsetzung im Rahmen des linguistischen Komponentensystems Tesla berücksichtigt. Wesentliches Ziel dieser Arbeit ist es, anhand konkreter computerlinguistischer Experimente zu zeigen, wie der dynamische Bedeutungsbegriff der Kognitiven Linguistik modelliert werden kann. Anhand einer Untersuchung mehrdeutiger sprachlicher Einheiten soll gezeigt werden, dass sich die Bedeutungskonstitution als ein Prozess der Herausbildung komplexer sprachlicher Muster erfassen lässt. Darauf aufbauend wird vom Phänomen der Mehrdeutigkeit abstrahiert, um die Hypothese zu prüfen, dass der Prozess einer kontextbedingten Bedeutungskonstitution ein allgemeines Prinzip ist, welches auch bei sprachlichen Einheiten mit einem vermeintlich eindeutigen Bedeutungspotential vorliegt.

Neben der experimentellen Erprobung des computerlinguistischen Modells wird damit auch der Zusammenhang zwischen Musterbildung und Bedeutungskonstitution zum Gegenstand der Arbeit. Die zentrale Annahme ist hierbei, dass sich Bedeutung durch eine Transformation von Verwendungsmustern modellieren lässt – in Abgrenzung zum WSM, bei dem das Verwendungsmuster selbst die Bedeutung repräsentiert. Damit verbunden ist auch die Frage nach den Bedingungen und Möglichkeiten eines rein datengetriebenen Ansatzes für das Problem der Bedeutungsermittlung; dies betrifft insbesondere die Frage nach der Eignung des WSM für die Modellierung eines dynamischen Bedeutungsbegriffs im Sinne der theoretischen Annahmen der Kognitiven Linguistik.

Das Wissen über sprachliche Systeme ist in dieser Arbeit somit nicht Voraussetzung, sondern Folge und Ergebnis der systemsprachlichen Analyse. Informationsverarbeitende Systeme sind in diesem Zusammenhang ein zentraler Bestandteil linguistischer Theoriebildung, insofern ihr Einsatz es ermöglicht, sprachliche

7 Auf das WSM wird häufig auch unter der Bezeichnung Distributional Semantic Models Bezug genommen, was auf die mit dem Modell oftmals assoziierte Idee einer Distributionellen Semantik verweist. Zu deren Verhältnis gegenüber dem in dieser Arbeit zugrunde gelegten Bedeutungsbegriff sei auf Kapitel 4 verwiesen.

Verwendungskontexte und -bedingungen unabhängig vom impliziten Vorwissen menschlicher Bearbeiter zu erschließen, um sie für eine systematische Analyse zugänglich zu machen (siehe dazu auch Rolshoven/Schwiebert 2007). Die Arbeit ist damit zwischen linguistischer Grundlagenforschung und computerlinguistischer Anwendung angesiedelt. Die eingesetzten Softwarewerkzeuge ermöglichen es zum einen, die linguistischen Hypothesen empirisch-experimentell zu überprüfen und mit Hilfe von Simulationen Einblick in die Dynamik sprachlicher Systeme zu geben, zum anderen können die Ergebnisse als Grundlage für Forschungsarbeiten und Anwendungen der maschinellen Sprachverarbeitung eingesetzt werden, etwa im Bereich des Text Mining oder des Information Retrieval. Mit der Formulierung konkreter Anwendungsfälle und der Bereitstellung der zugehörigen Verfahren und Ergebnisse über das in der Sprachlichen Informationsverarbeitung entwickelte Open-Source-Framework Tesla versteht sich die Dissertation somit auch als Beitrag zur Hervorhebung der Rolle computerlinguistischer Experimente für die sprachwissenschaftliche Theoriebildung.

1.2 Aufbau der Arbeit

Die Gliederung der Arbeit orientiert sich in wesentlichen Punkten an dem von David Marr (1982) vorgeschlagenen Vorgehen zur Beschreibung informationsverarbeitender Systeme.[8] Im Hinblick auf die Modellierung wird die Bedeutungskonstitution in dieser Arbeit im Sinne von Marr als ein informationsverarbeitender Prozess verstanden, bei dem im Wesentlichen eine Eingabeinformation in eine Ausgabeinformation überführt wird. Marr schlägt drei verschiedene Ebenen vor, anhand derer solche Prozesse in informationsverarbeitenden Systemen beschrieben werden können (Tabelle 1.1).

Computational theory	Representation and algorithm	Hardware implementation
What is the goal of the computation, why is it appropriate, and what is the logic of the strategy by which it can be carried out?	How can this computational theory be implemented? In particular, what is the representation for the input and output, and what is the algorithm for the transformation?	How can the representation and algorithm be realized physically?

Tabelle 1.1: Die drei Ebenen der Beschreibung nach Marr (1982), auf denen Systeme erfasst werden können, die informationsverarbeitende Prozesse ausführen (Tabelle übernommen aus Marr, 1982, 25).

8 In seinem für die Kognitionswissenschaft prägenden Buch »Vision« beschäftigt sich Marr mit Systemen der visuellen Wahrnehmung. Während Marr dabei in erster Linie auf die Analyse von komplexen Systemen zielt (er bezieht sich explizit auf »devices« bzw. »machines«), wird sein Vorgehen hier auf den sehr viel begrenzteren Fall der Beschreibung eines Modells der Bedeutungskonstitution übertragen.

Nach Marr muss auf der ersten Ebene zunächst eine Verarbeitungstheorie angegeben werden. Diese muss erklären, was das Ziel der Verarbeitung ist, wie sich das System in Bezug auf die Überführung von Input zu Output verhält und unter welchen Bedingungen diese Überführung geschieht. Auf der zweiten Ebene werden zum einen die Repräsentationen der Ein- und Ausgabeinformationen beschrieben, mit denen die Verarbeitung implementiert werden kann; zum anderen muss ein Algorithmus angegeben werden, der für die Transformation von Input zu Output zuständig ist. Dabei muss unter anderem auch thematisiert werden, welchen Einfluss die Wahl der Repräsentation auf den Algorithmus hat. Auf der dritten Ebene geht es schließlich um die tatsächliche (physische) Umsetzung des Modells, das heißt, wie ein solches System konkret realisiert werden kann.

Die mit diesen drei Beschreibungsebenen verbundenen Fragen bestimmen im Wesentlichen den Aufbau der Arbeit. Als Ausgangspunkt wird in Kapitel 2 das zu modellierende Phänomen beschrieben. Hierbei werden verschiedene Formen der Variabilität sprachlicher Bedeutung näher betrachtet; darauf aufbauend wird die Vorstellung eines flexiblen Bedeutungspotentials als übergreifende Problembeschreibung etabliert, woraus sich die Annahme eines dynamischen Bedeutungsbegriffs als notwendiges Desiderat herleiten lässt. Anschließend wird in Kapitel 3 in den Begriffen der Kognitiven Semantik eine Verarbeitungstheorie zur Erklärung des Phänomens der Bedeutungsvariation vorgeschlagen. Hierfür werden zunächst die grundlegenden Annahmen der Kognitiven Linguistik bzw. der Kognitiven Semantik dargelegt. Anschließend wird anhand des *dynamic construal approach* nach Cruse (2011); Croft/Cruse (2004) die Konzeption einer Bedeutungskonstitution in Relation zu einem allgemeinen Bedeutungspotential konkretisiert, welche die theoretische Grundlage für die computerlinguistische Modellierung in dieser Arbeit darstellt.

Die Leitfragen der zweiten Beschreibungsebene sind nach Marr, wie diese abstrakte Verarbeitungstheorie algorithmisch umgesetzt werden kann und welche Rolle die Wahl der Repräsentation dabei spielt. Da die Kognitive Semantik selbst keine klare Operationalisierung bereitstellt, wird in Kapitel 4 mit dem Word Space Model (WSM) ein bereits etabliertes computerlinguistisches Modell vorgestellt, das in dieser Arbeit als Grundlage sowohl für die Ermittlung und Repräsentation des Bedeutungspotentials als auch für die darauf aufsetzende Umsetzung der Bedeutungskonstitution dienen soll. Dabei muss vor allem auch das Verhältnis zum Bedeutungsbegriff der Kognitiven Linguistik diskutiert werden, da das WSM selbst zumeist mit einem gegenüber der Kognitiven Linguistik abweichenden, rein distributionellen Bedeutungsbegriff verknüpft wird. In Kapitel 5 wird daraufhin eine Operationalisierung der Bedeutungskonstitution mittels des WSM vorgeschlagen, welche sich als eine algorithmische Transformation der hier eingesetzten Repräsentationen beschreiben lässt. In einem weiteren Schritt wird gezeigt, wie die Ergebnisse der Transformation auch zueinander in Beziehung gesetzt werden können, um dadurch das tatsächliche Bedeutungspotential von Wörtern zu analysieren.

Gemäß Marrs Konzeption liegt die Perspektive in der dritten Beschreibungsebene auf der konkreten Realisierung des Modells, die in dieser Arbeit in Form einer softwaretechnologischen Umsetzung auf Grundlage des Text Engineering Software Laboratory (Tesla) erfolgt. In Tesla können die einzelnen Schritte der Operationalisierung in Komponenten gekapselt und innerhalb von verschiedenen experimentellen Konfigurationen in verschiedenen Konstellationen und Parametrisierungen eingesetzt werden. In Kapitel 6 wird zunächst das grundlegende experimentelle Setup in Tesla beschrieben. Neben einer Beschreibung der Daten werden hier auch die für die Modellierung notwendigen Verfahrensbestandteile charakterisiert und auf bereits vorhandene und im Zuge der Arbeit noch zu erstellende Komponenten abgebildet. Anschließend wird in Kapitel 7 die Anwendung des Modells in Form konkreter computerlinguistischer Experimente in Tesla beschrieben. Diese dienen im Sinne des obigen Zitats von Rieger der »experimentellen Erprobung« des Modells (Rieger 1985, 1), indem beispielhaft eine Auswahl mehrdeutiger Wörter in verschiedenen Kontextualisierungen verglichen wird.

In Kapitel 8 werden schließlich die Ergebnisse der Experimente vor dem Hintergrund der theoretischen Vorannahmen diskutiert, wobei vor allem auch auf notwendige Einschränkungen und Vereinfachungen gegenüber der theoretischen Konzeption von Cruse eingegangen wird. Die Arbeit schließt mit einer kritischen Bewertung der hier vorgeschlagenen Modellierung. Dabei wird insbesondere thematisiert, welche Rückschlüsse die Experimente auf die zugrunde gelegte theoretische Konzeption ermöglichen – und auch, was dies ganz allgemein für den Stellenwert einer computerlinguistischen Modellierung für kognitiv motivierte Theorien bedeutet.

2. Das Bedeutungspotential sprachlicher Einheiten

Gegenstand dieser Arbeit ist eine computerlinguistische Modellierung der Bedeutungskonstitution in sprachlichen Einheiten. Grundlage für diese Konzeption von Bedeutung ist die Annahme, dass sprachliche Einheiten über ein flexibles Bedeutungspotential verfügen, welches seinen Ausdruck in der hohen Variabilität sprachlicher Bedeutung findet. Bevor in Kapitel 3 die theoretische Konzeption der Bedeutungskonstitution aus Sicht der Kognitiven Semantik erörtert wird, um damit die Grundlage für die computerlinguistische Modellierung zu schaffen, soll in diesem Kapitel zunächst das Phänomen der Variabilität sprachlicher Bedeutung näher betrachtet werden, dessen Erklärung als das wesentliche Motiv der Kognitiven Semantik angesehen werden kann.

Hierfür werden in Abschnitt 2.1 zunächst verschiedene Arten der Mehrdeutigkeit aus Sicht der lexikalischen Semantik beschrieben. Neben einem Bedeutungswandel über Zeit und der Ambiguität sprachlicher Ausdrücke meint dies hier vor allem auch die allgemeine Bedeutungsvariation in Abhängigkeit vom Kontext, wie sie sich etwa in der sprechergebundenen Interpretation in verschiedenen Situationen zeigt. Auf dieser Grundlage wird in Abschnitt 2.2 das Phänomen der Variabilität sprachlicher Bedeutung als linguistisches Problem etabliert. Dabei wird der Begriff des Bedeutungspotentials als übergreifendes Konzept zur Beschreibung von Mehrdeutigkeiten eingeführt, sowie darauf aufbauend die Bedeutungskonstitution als notwendiges Desiderat eines dynamischen Bedeutungsbegriffs formuliert.

2.1 Die Variabilität sprachlicher Bedeutung

Ein offenkundiges Problem bei der Ermittlung und Darstellung sprachlicher Bedeutung ist deren Variabilität, ein allgegenwärtiges Phänomen, das in vielerlei Gestalt auftreten kann. Die Variabilität sprachlicher Bedeutung ist die Grundlage für Wortwitz und Pointe, für Missverständnis und Täuschung, für Metaphorik und Poesie. Ebenso vielfältig sind auch die Erscheinungsformen. Variabilität bezeichnet dabei zunächst einmal ganz allgemein den Umstand, dass sprachliche Ausdrücke oftmals mehrdeutig sind und dass sie deshalb auf mehrere, voneinander abweichende Arten interpretiert werden können. Hier muss zunächst unterschieden werden zwischen einer diachronen Perspektive, bei der Sprache über einen größeren Zeitraum hinweg untersucht wird, und einer synchronen Perspektive, bei der Sprache zu einem bestimmten Zeitpunkt betrachtet wird. In diachroner Perspektive ist die Bedeutung sprachlicher Ausdrücke in dem Sinne variabel, dass sie sich mit der Zeit verändern kann, ein gleiches Wort kann dabei

mitunter eine vollständig neue Bedeutung annehmen.[9] Dem Bedeutungswandel über einen bestimmten Zeitraum hinweg stehen bei einer synchronen Sprachbetrachtung vielfältige Erscheinungsformen von Mehrdeutigkeit gegenüber, von denen einige im Folgenden exemplarisch vorgestellt werden sollen.

In Abschnitt 2.1.1 wird das Phänomen der »Ambiguität« erläutert, welche in der lexikalischen Semantik als eine Eigenschaft angesehen wird, die nur bestimmten sprachlichen Ausdrücken zukommt, bei denen die Mehrdeutigkeit deshalb als ›lexikalisiert‹ angenommen wird. Anschließend werden weitere Formen der Mehrdeutigkeit thematisiert, die sich in einer als allgemeines Phänomen anzusehenden »Bedeutungsvariation« in verschiedenen (diskursiven) Kontexten äußern (Abschnitt 2.1.2).

2.1.1 Ambiguität

In der lexikalischen Semantik wird Mehrdeutigkeit unter dem Begriff der »Ambiguität« zusammengefasst. Diese wird in der Regel von der semantischen »Vagheit« abgegrenzt, welche die interpretatorische Unbestimmtheit hinsichtlich einiger weniger semantischer Merkmale bei einer festen Kernbedeutung bezeichnet, etwa bei Dimensionsadjektiven wie *groß*, *klein*, *hoch*, etc. Von Ambiguität wird in der lexikalischen Semantik immer dann gesprochen, wenn einem sprachlichen Ausdruck mehrere verschiedene Bedeutungen zugeordnet sind. Auf Ebene der Wortbedeutung wird dies unter dem Begriff der »lexikalischen Ambiguität« zusammengefasst. Im Wesentlichen wird hier zwischen zwei Unterarten unterschieden, die sich in verschiedenen semantischen Relationen äußern. »Homonymie« bezeichnet Fälle, in denen eine einzige Wortform mit zwei oder mehreren voneinander unabhängigen Bedeutungen assoziiert ist, die eine abweichende Etymologie aufweisen. Ein typisches Beispiel für Homonymie ist das Wort *Schloss*, welches wie in Beispiel 2.1 sowohl ein Gebäude als auch eine Schließvorrichtung bezeichnen kann:

Beispiel 2.1
a. Ein Schloss besichtigen
b. Ein Schloss aufbrechen

In der Lexikographie wird hier auch von »Homographie« gesprochen, da es sich im eigentlichen Sinne um eine gleiche graphematische Erscheinung zweier unterschiedlicher Wörter handelt, was sich in der lexikographischen Praxis in (mindestens) zwei Haupteinträgen niederschlägt.

9 So hat z.B. das Wort *Gesindel* heute eine vollkommen andere Bedeutung als noch im vorvergangenen Jahrhundert. Meinte dies damals schlicht die einfachen Bediensteten in der Land- und Hauswirtschaft, so wird diese Bezeichnung heute vor allem abwertend verwendet. Die Veränderung der Bedeutung geht dabei oftmals mit den vielfältigen Veränderungen in Alltag, Lebensweise und -gewohnheiten einher. Die verschiedenen Bedeutungen bleiben dabei eine Zeitlang nebeneinander bestehen.

»Polysemie« dagegen liegt vor, wenn die verschiedenen, einem gleichen Ausdruck zugeordneten Bedeutungen einen gemeinsamen Bedeutungskern haben. Die verschiedenen (Teil-)Bedeutungen stehen dabei in enger Beziehung zueinander, insofern sie über eine gemeinsame Kernbedeutung verfügen. Ein vielzitiertes Beispiel für die sogenannte »systematische Polysemie« findet sich in Bierwisch (1983, 77): So kann das Wort *Schule* in einer Vielzahl verschiedener Lesarten verwendet werden, etwa als Institution, als Gebäude oder als Beschäftigungsart. Gerade für Verben ist eine solche systematische Polysemie als Normalfall anzusehen, da Verben in der Regel komplexe Handlungszusammenhänge beschreiben und ihre Eindeutigkeit meist erst durch den Gegenstand der Handlung ausreichend charakterisiert wird. So werden etwa in Beispiel 2.2 trotz gleichlautendem Verb zwei verschiedene Handlungen beschrieben:

Beispiel 2.2
a. Klavier spielen
b. Fußball spielen

Eine weitere Form von Mehrdeutigkeit beschreibt Cruse (1986, 66) in Abgrenzung zu rein syntaktischer Ambiguität[10] als »lexiko-syntaktisch«, wie sie etwa in Beispiel 2.3 vorliegt. Die lexiko-syntaktische Ambiguität bezeichnet Fälle, in denen eine gleiche Wortform unterschiedlichen syntaktischen Kategorien zugeordnet werden kann.

Beispiel 2.3
a. Das kommt mir sehr gelegen (A)
b. Wir haben im Urlaub jeden Tag in der Sonne gelegen (V)

Solche Fälle sind unter anderem im Englischen weit verbreitet, da hier viele Verben in gleicher graphematischer Erscheinung auch als Substantiv (*to work – the work; to run – the run* etc.) oder als Adjektiv auftreten können (wie zum Beispiel in dem Satz »I saw the door open.«).[11] Wie Beispiel 2.4 zeigt, ist eine solche Substantivierung auch im Deutschen nicht unüblich:

Beispiel 2.4
a. das Essen – etwas essen
b. die Arbeiten – lange arbeiten

10 Ein klassisches Beispiel für syntaktische Ambiguität ist der Satz »Ich sehe den Mann mit dem Fernglas«. Die Ambiguität resultiert hier aus der möglichen Zuschreibung alternativer Konstituentenstrukturen: *mit dem Fernglas* kann als modifizierende Präpositionalphrase (PP) sowohl der Nominalphrase (NP) *den Mann* als auch der Verbalphrase (VP) *sehe* zugeordnet werden. Dieses Problem ist nicht lexikalischen Ursprungs und ist deshalb nicht auf die Bedeutung der einzelnen Wörter zurückzuführen (siehe dazu Cruse 1986, 66). Dass derartige Beispielsätze nicht völlig aus der Luft gegriffen sind, beweist die folgende Schlagzeile, gefunden auf ZEIT Online am 24.11.2014: »Kind mit Spielzeugwaffe von Polizei erschossen« (siehe http://www.zeit.de/gesellschaft/zeitgeschehen/2014-11/cleveland-usa-polizei-erschiesst-zwoelfjaehrigen – Zugriff vom 21.02.2018).

11 Beispiel übernommen aus (Cruse 1986, 66).

Jedoch wird die Ambiguität hier in der Regel – zumindest bei einer graphematischen Betrachtung – durch die Normen der Groß-/Kleinschreibung weitgehend eingeschränkt. Häufiger anzutreffen sind im Deutschen die in den Beispielen 2.5 und 2.6 dargestellten Fälle einer adverbialen Verwendung von Partizipien:

Beispiel 2.5 a. die Haare gefärbt tragen (A)
b. sie hat sich die Haare gefärbt (V)

Beispiel 2.6 a. etwas gekühlt servieren (A)
b. Hast du die Getränke gekühlt?

Den hier beschriebenen Formen von Ambiguität ist gemeinsam, dass ihre jeweiligen (Teil-)Bedeutungen in der Regel separat lexikalisiert werden. Unterstützt wird dies durch Evidenz aus der sprachübergreifenden Betrachtung: So werden die verschiedenen Lesarten einer ambigen Wortform in anderen Sprachen oftmals unterschiedlich übersetzt. Grundlage ist dabei offenbar die Bedeutung und nicht die phonologische bzw. graphematische Form. Besonders deutlich wird dies bei Homonymen. Da Homonymie ein rein akzidentielles und damit sprachspezifisches Phänomen ist, bei dem konkurrierende Bedeutungen mit abweichender Etymologie einer gleichen Wortform zugeordnet sind, setzt sich in Beispiel 2.7 die im Deutschen für das Wort Bank vorliegende Homonymie im Englischen genauso wenig fort wie im umgekehrten Falle in Beispiel 2.8.

Beispiel 2.7 a. Bank – *bank* (Geldinstitut)
b. Bank – *bench* (Sitzbank)

Beispiel 2.8 a. *bank* – Ufer
b. *bank* – Bank (Geldinstitut)

Bei polysemen Wörtern lässt sich in der sprachübergreifenden Betrachtung ebenfalls oftmals eine lexikalische Abweichung feststellen (Beispiel 2.9):

Beispiel 2.9 a. Flügel – *wing*
b. Flügel – *grand piano*

Und auch bei der lexiko-syntaktischen Ambiguität in Beispiel 2.10 erfolgt eine mehrfache Lexikalisierung in der Zielsprache, da es sich hier aufgrund der Zugehörigkeit zu unterschiedlichen syntaktischen Kategorien ebenfalls um zwei verschiedene Wörter handelt:

Beispiel 2.10 a. *swallow* (V) – schlucken
b. *swallow* (N) – Schwalbe

Die wesentliche Gemeinsamkeit der verschiedenen Formen von Mehrdeutigkeit besteht somit in einer mehrfachen Lexikalisierung. Ambiguität wird in der lexikalischen Semantik als eine spezifische Eigenschaft bestimmter Wörter verstanden und stellt demnach eher eine Ausnahme von der Regel dar. Anders verhält es sich dagegen bei dem Phänomen der Bedeutungsvariation, um das es im Folgenden gehen soll.

2.1.2 Bedeutungsvariation

Die Variabilität von Bedeutung wurde zu Beginn dieses Kapitels als ein allgegenwärtiges Phänomen bezeichnet, und tatsächlich macht die im vergangenen Abschnitt beschriebene Mehrdeutigkeit im Sinne von lexikalischer Ambiguität nur einen kleinen Anteil der möglichen Ausprägungen aus. Im Allgemeinen äußert sich Mehrdeutigkeit vor allem in einer Variation der Bedeutung in Abhängigkeit vom jeweiligen Kontext. Diese bezeichnet den Umstand, dass sprachliche Ausdrücke in verschiedener Verwendung zum Teil erhebliche Bedeutungsveränderungen erfahren können, ohne dass sie deshalb im obigen Sinne als lexikalisch ambig bezeichnet werden müssten. So lassen sich etwa in Beispiel 2.11 für die einzelnen Verwendungen nicht zwingend einzelne Lesarten abgrenzen:

Beispiel 2.11
a. Zum Meer läuft man keine zwei Minuten.
b. Die Fähre läuft gerade ein.
c. Er läuft in die Küche.
d. Er läuft jeden morgen eine halbe Stunde.
e. Diese Unterscheidung läuft ins Leere.

Unabhängig von einer detaillierten Analyse ist hier entscheidend, dass die Variation der Bedeutung offenkundig auch mit Unterschieden in den konkreten Kontextualisierungen einhergeht. Die Bedeutungsvariation spiegelt sich demnach vor allem auch in den unterschiedlichen Gebrauchskontexten wider.[12] Sprachliche Ausdrücke werden somit in gewissem Sinne stets durch andere sprachliche Ausdrücke beschrieben bzw. spezifiziert. Eingebettet in einen konkreten Kontext fällt die Interpretation in der Regel nicht schwer – zumeist selbst dann nicht, wenn es sich um einen ambigen Ausdruck handelt. In diesem Sinne ist Mehrdeutigkeit

12 Darüber hinaus gestatten sprachliche Ausdrücke ganz grundsätzlich die Möglichkeit zu einer unterschiedlichen Interpretation durch verschiedene Sprecher bzw. Sprechergruppen. Dies wird insbesondere im Falle von Umgangssprache deutlich, die ganz allgemein als eine Abweichung von sprachlichen Normen angesehen werden kann, u.a. eine regionale (etwa bei Dialekten bzw. Regiolekten) oder auch soziale Komponente (bei sogenannten Soziolekten) widerspiegeln, etwa die Anbindung an bestimmte Milieus (so basieren beispielsweise Jugendsprachen auf dieser Art der Distinktion). Die Bedeutung (bzw. deren Interpretation) hängt dabei einerseits vom jeweiligen Sprecher, andererseits aber auch von den jeweiligen situativen, sozialen, regionalen oder auch historischen Kontexten ab, in denen ein sprachlicher Ausdruck auftritt.

nicht etwa eine besondere Eigenschaft, die nur bestimmten sprachlichen Einheiten zukommt, sondern ein allgemeines Phänomen, das in der konkreten Verwendung in den Hintergrund tritt.[13]

Die tragende Rolle des Kontextes ist in der lexikalischen Semantik heute weitgehend unbestritten.[14] Tatsächlich existiert eine ganze Reihe von gebrauchsorientierten Ansätzen, die die konkreten Verwendungsmuster in den Mittelpunkt stellen und daher oftmals als »usage-based« bezeichnet werden. Die Strategien für die Einbindung kontextueller Informationen fallen dabei jedoch höchst unterschiedlich aus. Die Varianten der Kontextualisierung reichen von der Definition syntagmatischer Affinitäten, die die Selektion von (Teil-)Bedeutungen durch den Kontext steuern (siehe dazu Cruse 1986) über die Einarbeitung generischer Gebrauchskontexte in semantisch orientierte Lexika in der Generative Lexicon Theory nach Pustejovsky (1998) bis hin zur dynamischen Bedeutungskonzeption der Kognitiven Semantik, welche im nachfolgenden Kapitel näher betrachtet wird.

2.2 Zusammenfassung

In diesem Kapitel wurden – ohne jeden Anspruch auf Vollständigkeit – verschiedene Erscheinungsformen der Variabilität sprachlicher Bedeutung vorgestellt, wobei im Wesentlichen zwischen lexikalischer Ambiguität und einer Bedeutungsvariation im Kontext unterschieden wurde. Unabhängig von einer genauen Typologie verschiedener Formen von Mehrdeutigkeit lässt sich zunächst festhalten, dass es sich hierbei nicht um ein sporadisch auftretendes Phänomen handelt, sondern dass Mehrdeutigkeit, insbesondere im Sinne einer Bedeutungsvariation, vielmehr die Regel ist. Für eine übergreifende Beschreibung der Variabilität sprachlicher Bedeutung bietet sich hier der Begriff des »Bedeutungspotentials«[15] an: sprachliche Einheiten verfügen über ein flexibles Bedeutungspotential, welches ihnen ermöglicht, in verschiedenen Kontexten verschiedene Bedeutungen einzubringen. Dass dies in der Kommunikation dennoch nicht permanent zu Unverständlichkeit führt, ist vor allem dem hohen Grad an Selbstreflexivität von Sprache zu verdanken. So ist das Sprechen über Sprache nicht nur ein wichtiger Bestandteil der Arbeit von Linguisten, sondern auch ein wesentliches Merkmal der alltäglichen

13 Abgesehen von ihrem bewussten Einsatz, wie er bspw. in den zu Beginn des Kapitels genannten Formen vorliegt, also etwa in Wortwitz, Pointe, Poesie etc.

14 In der lexikalischen Semantik hat sich die Sicht einer kontextbasierten Konzeption von Bedeutung erst im Laufe des 20. Jahrhunderts gegen die Vorstellung von Bedeutungen als weitgehend statische Objekte im (mentalen) Lexikon durchgesetzt (siehe dazu z.B. Zlatev 2003).

15 Der hier verwendete Begriff des Bedeutungspotentials beschränkt sich auf die Ebene der Wortbedeutung. Er unterscheidet sich damit u.a. von dem durch Michael Halliday geprägten Begriff des »meaning potential« (vgl. Halliday 1973; 1978). Bei Halliday ist es die Sprache als Ganzes, die über ein Bedeutungspotential verfügt, im Sinne von einem ›Potential, zu bedeuten‹; bei Halliday bezeichnet das »meaning potential« somit eine grundlegende Eigenschaft des gesamten Sprachsystems.

Kommunikation – hier wird die genaue Bedeutung von Aussagen im Falle von Unklarheiten mittels Ergänzung und Paraphrasierung fokussiert.[16] Nicht zuletzt darin liegt auch begründet, dass Ambiguitäten im konkreten Sprachgebrauch nur selten eine ernsthafte Rolle spielen.

Während sich das Phänomen im Diskurs somit eher als Ausnahme darstellt, etabliert sich Ambiguität offenbar immer dann als linguistisches Problem, wenn bei der Betrachtung sprachlicher Einheiten von einer konkreten Verwendung abstrahiert wird. Eine solche Abstraktion wird beispielsweise in Lexika vollzogen, in denen sprachliche Einheiten zumeist als isolierte Einträge gelistet werden. Das Problem der isolierten Betrachtung tritt jedoch auch und vor allem in sprachtechnologischer Perspektive deutlich hervor, wie sich etwa am Beispiel von Suchmaschinen illustrieren lässt: Wo Suchmaschinen mitunter mit einem suggestiven »Meinten Sie…?« reagieren, haben Sprecher in der Regel keine Probleme, die jeweilige Bedeutung zu erfassen.[17]

Den verschiedenen Formen von Mehrdeutigkeit ist gemeinsam, dass sie vor allem an lokal isolierten Stellen auftreten (wie dies beispielsweise in Lexika oder Suchmaschinen gegeben ist): In isolierter Betrachtung sind sprachliche Ausdrücke hinsichtlich ihrer Bedeutung unbestimmt. Unbestimmtheit ist damit ein allgemeinerer Begriff, um Mehrdeutigkeit zu beschreiben: Er besagt, dass sprachliche Ausdrücke in isolierter Betrachtung ›unterspezifiziert‹ sind. Diese Unterspezifiziertheit muss jedoch nicht als Mangel ausgelegt werden,[18] sondern kann vielmehr als Ausdruck der semantischen Dynamik von Sprache verstanden werden, welche die variable Verwendung sprachlicher Einheiten ermöglicht. In dieser Perspektive ist Bedeutungsvariation in erster Linie Ausdruck des hohen Maßes an Ökonomie, über das natürliche Sprache verfügt: Die Mehrdeutigkeit sprachlicher Ausdrücke ermöglicht den flexiblen Einsatz eines begrenzten Zeicheninventars für verschiedene kommunikative Ziele.

Mit der Annahme eines flexiblen Bedeutungspotentials entsteht gleichsam ein Desiderat: Zwar lässt sich dadurch erklären, warum ein sprachlicher Ausdruck mehrere Bedeutungen haben kann; es lässt für sich genommen jedoch offen, wie die Variation der Bedeutung in verschiedenen Kontexten begründet ist. Für ein vollständiges Bild fehlt noch ein Mechanismus bzw. ein Prozess, der eine Erklärung dafür bietet, warum in verschiedenen Kontexten verschiedene Bedeutungen auftreten können und warum in einem konkreten Kontext scheinbar dennoch

16 Hinzu kommt eine Vielzahl zusätzlicher Informationen wie der situative Kontext, Hintergrundwissen, etc., auf die im Rahmen dieser Arbeit jedoch nicht eingegangen werden kann.

17 Suchmaschinen stehen für ihren speziellen Anwendungsbereich eine Vielzahl hervorragender Strategien zur Verfügung, etwa die Einbeziehung von Browserprofilen, Suchverlauf, Ranking etc. Aus sprachtheoretischer Sicht verfügen diese Strategien jedoch zumeist nur über ein relativ geringes explanatorisches Potential.

18 Vgl. bspw. formalsemantische Ansätze, deren Sicht impliziert, dass Mehrdeutigkeit ein Mangel ist, den es mittels einer formalen Analyse auszugleichen gilt.

zumeist nur eine dieser Bedeutungen vorliegt,[19] bzw. – in der hier gewählten Terminologie – warum jeweils nur Teile des Bedeutungspotentials zum Tragen kommen. Dieser Prozess lässt sich in der Differenz von allgemeinem Bedeutungspotential und konkreter, kontextualisierter Bedeutung verorten: In isolierter Betrachtung bleibt die Bedeutung unbestimmt, und erst durch die Einbettung in einen konkreten Kontext wird diese Unbestimmtheit aufgehoben.

Rieger (1985) stellt hier die Forderung nach einer prozeduralen Semantik auf, bei der Bedeutung nicht als dauerhaft bzw. statisch verstanden wird, sondern vielmehr als ein kontinuierlicher Prozess: Bedeutung ›konstituiert‹ sich erst im konkreten Kontext, und diese Bedeutungskonstitution ist ein ›lokaler‹ Prozess. Über das Konzept der Bedeutungskonstitution können nach Rieger »Phänomene wie Variabilität, Vagheit, Vorläufigkeit, Revidierbarkeit, [sic!] etc. […] in die Untersuchungen einbezogen werden, und zwar nicht als Defizite [sic!] sondern als erklärte Resultate der Dynamik Bedeutung konstituierender Prozesse« (Rieger 1985, 9). Diese radikale, heute insbesondere auch in der Kognitiven Linguistik verbreitete und dort unter anderem von Alan Cruse (siehe etwa Cruse 2011; Croft/Cruse 2004) vertretene Sicht, dass Wortbedeutungen nicht für sich existieren, sondern sich jeweils nur *online*, das heißt im Zuge der konkreten Verwendung konstituieren, läuft letztlich darauf hinaus, dass die Bedeutung in geradezu ›jedem‹ Kontext ein bisschen variiert.

Angesichts eines solch dynamischen, rein kontextbasierten Bedeutungsbegriffs ergeben sich im Hinblick auf eine Modellierung im Wesentlichen zwei Teilprobleme: Zum einen die Frage, wie das Bedeutungspotential repräsentiert sein muss, damit sich daraus verschiedene (Teil-)Bedeutungen ableiten lassen. Zum anderen die Frage nach dem Prozess der Ableitung selbst, das heißt, welche Faktoren den Prozess anstoßen und welche Rolle diese Faktoren in einem solchen Prozess einnehmen. Diese Fragen stehen im Mittelpunkt dieser Arbeit. Im folgenden Kapitel wird zunächst die Position der Kognitiven Semantik anhand der in diesem Kapitel eingeführten Begriffe des Bedeutungspotentials und der Bedeutungskonstitution herausgearbeitet. Aus Sicht der Computerlinguistik stellt sich im Anschluss vor allem die Frage nach einer angemessenen Modellierung, die eine empirische Überprüfung der theoretischen Annahmen ermöglicht.

19 Es gibt hier, wie oben angedeutet, selbstverständlich eine Reihe von Ausnahmen, die jedoch weniger in den Bereich der lexikalischen Semantik als vielmehr in die Pragmatik fallen.

3. Bedeutungspotential und Bedeutungskonstitution

In diesem Kapitel werden die für die Modellierung maßgeblichen Konzepte erörtert, namentlich das Bedeutungspotential sprachlicher Einheiten und die Bedeutungskonstitution in konkreten sprachlichen Kontexten. Diese Begriffe verweisen auf den konzeptuellen Bezugsrahmen der Kognitiven Semantik. Diese ist ein Teilgebiet der Kognitiven Linguistik, deren grundsätzliche Positionen zunächst in Abschnitt 3.1 dargestellt werden. Auf dieser Grundlage wird in Abschnitt 3.2 zum einen der Begriff des Bedeutungspotentials als eines der zentralen Konzepte der Kognitiven Semantik erörtert, zum anderen wird mittels Cruses *dynamic construal approach* die Bedeutungskonstitution beschrieben. Als ein erster Schritt hin zur Modellierung wird abschließend ein schematisches Modell der Bedeutungskonstitution skizziert, das den Ausgangspunkt für die spätere Operationalisierung der Bedeutungskonstitution bildet. Die theoretische Konzeption wird schließlich in Abschnitt 3.3 noch einmal zusammengefasst und eingeordnet. Hierbei wird vor allem eine Eingrenzung vorgenommen, da nicht die Kognitive Linguistik als Ganzes modelliert wird, sondern mit der Bedeutungskonstitution nur ein spezifischer Teilaspekt.

3.1 Kognitive Linguistik

Die Kognitive Linguistik versteht sich als ein Teilgebiet der interdisziplinär ausgerichteten Kognitionswissenschaften, welche unter anderem Einflüsse aus der Kognitiven Psychologie, der Künstlichen Intelligenz, der Kognitiven Neurowissenschaften und der Linguistik zu einem gemeinsamen Forschungsrahmen vereinen. Die Ursprünge der Kognitiven Linguistik werden zumeist in den späten 1950er Jahren verortet. Rickheit u.a. (2010, 10) sprechen hier von einer »kognitiven Wende« (siehe dazu auch Schwarz 2008, 15f.), die im Wesentlichen durch die Arbeiten von Noam Chomsky (1957, 1965) markiert ist.[20] Linguistische Theorien sind demnach seit Chomsky insofern als kognitiv gekennzeichnet, als hier eine Abgrenzung zum behaviouristischen Ansatz einer positivistischen Beschränkung auf »beobachtbare Phänomene« vorgenommen wird (siehe Schwarz 2008, 15). Chomsky setzte dem die These einer angeborenen, genetisch determinierten Sprachfähigkeit entgegen, die sich in einem autonomen kognitiven Sprachmodul manifestiert. Den Kern dieser angeborenen Sprachfähigkeit bilden grammatische Regeln, welche nach Chomsky die Grundlage der sprachlichen Generativität ausmachen, also »der

20 Als besonders folgenreich erwies sich Chomskys 1959 veröffentlichte Kritik an B. F. Skinners Buch »Verbal Behaviour« (Skinner 1957), in der u.a. das Argument des »poverty of stimulus« formuliert ist. Dessen Kernaussage, dass der kindliche Spracherwerb unmöglich allein auf sprachlichem Input in Verbindung mit Reiz-Reaktions-Schemata basieren könne, schien mit einem Mal die zu jener Zeit vorherrschenden, empirisch ausgerichteten Ansätze insgesamt zu widerlegen.

Fähigkeit zur Bildung unendlich vieler grammatisch korrekter Sätze« (Schwarz 2008, 15) aus einem begrenzten Zeicheninventar.

Ziel der durch Chomsky begründeten Generativen Grammatik war die Suche nach einer Universalgrammatik (UG), die als grundlegend und überindividuell verstanden wird. Mit der Annahme eines idealisierten Sprecher-Hörers stand dabei – anders als etwa in früheren strukturalistischen Ansätzen – »nicht mehr das konkrete Verhalten (in Chomskys Terminologie: die Performanz) im Mittelpunkt sprachwissenschaftlicher Untersuchungen, sondern das diesem Verhalten zugrundeliegende Kenntnissystem (die Kompetenz)« (siehe Schwarz 2008, 17). Vor allem diese strikte Trennung von Kompetenz und Performanz, die aus dem Strukturalismus übernommen wurde,[21] hat die rationalistische Ausrichtung der Generativen Grammatik nachhaltig bestärkt: Sie war das Fundament, auf dem begründet werden konnte, warum die Performanz, also die ›Sprache im Gebrauch‹, als zweitrangig angesehen werden kann. Die Generative Grammatik konzentrierte sich fortan auf die Syntax, die weitgehend mit der Kompetenz identifiziert wurde. In der Performanz begründete semantische Phänomene wurden hingegen weitgehend in das Lexikon ausgelagert und ihre Rolle für die Theoriebildung damit marginalisiert.

Nachdem sich mit Chomskys biologistischer Konzeption einer angeborenen Sprachfähigkeit zunächst »[w]issenschaftstheoretisch und -historisch […] die Wende vom ›reinen‹ Empirismus zum ›reinen‹ Rationalismus in der herrschenden Lehre der Linguistik« vollzogen hatte (siehe Rickheit u.a. 2010, 10), wurde der damit vorwiegend rationalistische Weg zur Erkenntnisgewinnung mit dem verstärktem Aufkommen psycholinguistischer Ansätze in den 1970er Jahren wieder zunehmend durch empirische Untersuchungen ergänzt. So wurden die syntaktisch orientierten, auf die Kompetenz ausgerichteten Ansätze in der Tradition der Generativen Grammatik um stärker semantikorientierte, auf die Performanz ausgerichtete Ansätze erweitert. Im Zuge dessen wurde die Vorstellung eines autonomen Sprachmoduls nach und nach abgelöst durch die Vorstellung, dass die Sprachfähigkeit nicht unabhängig von anderen kognitiven Fähigkeiten gesehen werden kann, sondern dass sie mindestens mit anderen kognitiven Fähigkeiten interagiert – oder gar vollständig in diesen begründet ist.

Bei diesen neueren, stärker empirisch ausgerichteten Ansätzen kann nach Schwarz (2008, 48f.) grundsätzlich zwischen zwei Positionen unterschieden werden: Dem modularen und dem holistischen Ansatz. Der modulare Ansatz hat seinen Ursprung in der syntaktisch orientierten Generativen Grammatik bzw. war nach Schwarz lange Zeit eng mit dieser verbunden. Mit dem Generativismus teilt der modulare Ansatz die Annahme, dass es ein spezifisches Sprachmodul

21 Wobei die von Ferdinand de Saussure im »Cours de linguistique générale« (Saussure 1967) getroffene Unterscheidung zwischen »langue« und »parole« nicht unmittelbar gleichzusetzen ist; so verfügt bspw. die »langue« in ihrer ursprünglichen Konzeption im Gegensatz zu Chomskys »Kompetenz« über eine soziale Dimension (vgl. dazu Wunderli 2014, 185f.).

gibt, er sieht dieses jedoch als nicht vollständig autonom, sondern begreift es in Interaktion zu anderen Modulen der Kognition, die Schwarz als verschiedene »Kenntnissysteme« bezeichnet (Schwarz 2008, 49).[22] Diese umfassen neben sprachlichem Wissen unter anderem auch Weltwissen und das Wissen über soziale Situationen. Neben der Unterscheidung verschiedener Wissensformen und deren Repräsentation sieht der modulare Ansatz auch nach wie vor eine Trennung von linguistischen und nicht linguistischen kognitiven Fähigkeiten vor. Der holistische Ansatz hingegen verneint ebendiese Trennung und sieht Sprache bzw. die Sprachfähigkeit im Allgemeinen als Ausdruck allgemeiner kognitiver Prinzipien an.

Die Unterscheidung zwischen modularem und holistischem Ansatz findet im englischen Sprachraum auch eine orthographische Entsprechung: Während mit der kleingeschriebenen Variante (*cognitive linguistics*) in der Regel der modulare Ansatz bezeichnet wird, ist in der großgeschriebenen (*Cognitive Linguistics*) der holistische Ansatz gemeint.[23] Letzterer bildet den Bezugsrahmen dieser Arbeit – wenn im Folgenden von »Kognitiver Linguistik« die Rede ist, so ist damit der holistische Ansatz gemeint.

3.1.1 Holistischer Ansatz

Die holistisch ausgerichtete Kognitive Linguistik hat sich seit etwa Mitte der 1970er Jahre vor allem im englischen Sprachraum als eigenständige Forschungsrichtung etabliert. Maßgebliche Arbeiten sind unter anderem die Frame-Semantik nach Charles Fillmore (1976; 1982), George Lakoffs Arbeiten zu Metaphern (Lakoff/Johnson 1980) und zur Kategorisierung (Lakoff 1987) sowie insbesondere auch Ronald Langackers Konzeption einer *cognitive grammar* (Langacker 1987; 1991). Einige grundlegende Texte sind in Geeraerts (2006a) zusammengefasst; einen sehr guten Überblick geben zudem Evans/Green (2006) sowie vor allem Croft/Cruse (2004), in dem die Autoren auf Grundlage der oben genannten Arbeiten zudem einen eigenen Ansatz entwickeln.

22 Schwarz bezieht sich in ihrer Darstellung des modularen Ansatzes v. a. auf Bierwisch, dessen Zwei-Ebenen-Semantik eine Trennung von semantischer und konzeptueller Repräsentationsebene vornimmt und diese als zwei unterschiedliche Module der Kognition ansieht (siehe dazu Bierwisch/Lang 1987; Lang/Bierwisch 1989).

23 Im Deutschen fehlt diese orthographische Unterscheidung, was die Ambiguität der Bezeichnung »Kognitive Linguistik« noch verstärkt und nach Schwarz (2008, 56f.) mitunter dazu führt, dass die Verschiedenheit der durch den Terminus bezeichneten Richtungen nicht wahrgenommen wird. Während die (kleingeschriebene) kognitive Linguistik die Gesamtheit der Ansätze bezeichnet, die Sprache als mentales Phänomen begreifen, also z.B. auch die Generative Grammatik, bezieht sich die (großgeschriebene) Kognitive Linguistik nur auf eine Teilmenge dieser Ansätze; und zwar auf jene, die sich nicht nur in Abgrenzung zu nicht kognitiv ausgerichteten Ansätzen verstehen, sondern vielmehr ganz klar Stellung beziehen gegen alle Ansätze, die Sprache als eine autonome kognitive Fähigkeit bzw. als durch ein spezifisches Sprachmodul realisiert ansehen – selbst wenn dieses, wie etwa bei Schwarz (bzw. allgemein im modularen Ansatz), als integriert in das kognitive System angesehen wird.

Nach Croft/Cruse (2004) versteht sich die Kognitive Linguistik als Gegenbewegung zur Generativen Grammatik sowie zu den in ihrem Umfeld entstandenen formalsemantischen Ansätzen, wie zum Beispiel der Montague Grammar (siehe Thomason 1974). Die Ablehnung eines unabhängigen Sprachmoduls, die hier als wesentliches abgrenzendes Kriterium für die Unterscheidung von holistischem und modularem Ansatz innerhalb der Kognitiven Linguistik verwendet wird, ist gleichzeitig die erste von drei grundlegenden Thesen, mit denen Croft/Cruse (2004, 1) die Kognitive Linguistik charakterisieren:

1. language is not an autonomous cognitive faculty
2. grammar is conceptualization
3. knowledge of language emerges from language use

Die erste These wendet sich explizit gegen die lange Zeit vorherrschende Generative Grammatik. Mit der Abgrenzung zu deren Autonomiehypothese lehnen Croft und Cruse insbesondere die Vorstellung ab, dass zwischen linguistischem und konzeptuellem Wissen unterschieden werden kann. Vielmehr geht die Kognitive Linguistik davon aus, dass linguistisches Wissen auf die gleiche Art und Weise repräsentiert ist wie andere Arten konzeptueller Strukturen auch – mit anderen Worten, dass linguistisches Wissen konzeptuelles Wissen ›ist‹. Gleichsam ist auch bezüglich der kognitiven Prozesse, die dieses Wissen involvieren, keine Unterscheidung zwischen linguistischen und nicht linguistischen kognitiven Fähigkeiten möglich. Anders als der modulare Ansatz sieht die Kognitive Linguistik holistischer Prägung Sprache bzw. sprachliche Phänomene demnach als Ausdruck der allgemeinen kognitiven Prinzipien und Fähigkeiten an. Sprache und Sprachfähigkeit werden somit nicht als autonom bzw. gleichwie isolierbar angesehen, vielmehr sind »Sprachfähigkeit und allgemeine kognitive Fähigkeiten […] in diesem Ansatz untrennbar miteinander verbunden« (siehe Schwarz 2008, 54).

Doch welches sind diese »allgemeinen kognitiven Fähigkeiten«, die anstelle eines autonomen Sprachmoduls angenommen werden? Nach Schwarz ist es »Ziel der holistisch ausgerichteten Kognitionsforschung […], die Menge der universalen Prinzipien (wie Konzeptualisierung, Mustererkennung, Kategorisierung usw.) zu beschreiben, die allen mentalen Fähigkeiten gleichermaßen zugrundeliegen« (siehe Schwarz 2008, 54), also auch den sprachlichen Fähigkeiten. Eine zentrale Rolle, insbesondere in Bezug auf sprachliche Bedeutung, spielt dabei die Konzeptualisierung, welche Gegenstand der oben genannten zweiten Hypothese ist. Unter Verweis auf Langacker nehmen Croft/Cruse an, »[…] that all aspects of conceptual structure are subject to construal« (siehe Croft/Cruse 2004, 3), dass also konzeptuelle Strukturen grundsätzlich Konzeptualisierungen im Sinne sogenannter *construals*[24] involvieren, sowohl hinsichtlich der zu kommunizierenden

24 Der von Langacker übernommene Begriff des *construal* lässt sich in etwa mit ›Deutung‹ bzw. ›Auslegung‹ übersetzen und bezeichnet hier die Art und Weise, wie ein bestimmter Begriff bzw.

Erfahrungen als auch in Bezug auf das zugrundeliegende linguistische Wissen, über das wir verfügen. Mit der Heraushebung der Rolle der Konzeptualisierung für den Aufbau von und die Bezugnahme auf konzeptuelle Strukturen wendet sich die These dezidiert gegen die Tradition einer wahrheitskonditionalen Semantik: konzeptuelle Strukturen können aus Sicht der Kognitiven Linguistik nicht auf eine wahrheitskonditionale Beziehung zur Welt reduziert werden, sprachliche Bedeutungen können vielmehr nur über die mit ihnen einhergehenden Konzeptualisierungen erschlossen werden.

Mit ihrer dritten These, der zufolge das Wissen über Sprache aus dem Sprachgebrauch emergiert,[25] wenden sich Croft/Cruse schließlich gegen die Tendenz zur Reduktion auf »maximally abstract and general representations of grammatical form and meaning« (Croft/Cruse 2004, 4), von der sowohl die Generative Grammatik als auch die wahrheitskonditionale Semantik geprägt ist. Während dort das (angeborene) linguistische Wissen in Form stark generalisierter und abstrakter Regeln den Sprachgebrauch bestimmt, nimmt die holistische Kognitive Linguistik hier eine Wechselbeziehung an: Die Regeln und Regularitäten der Sprachverwendung gehen demnach unmittelbar aus dieser hervor bzw. werden durch die Kognition mittels Konzeptualisierung aus konkreten Verwendungen abstrahiert. Emergenz von linguistischem Wissen erfolgt hier durch einen induktiven Prozess der Abstraktion, bei dem die »conventionalized subtleties and differences found among even highly specific grammatical constructions and word meanings« nicht verloren gehen (Croft/Cruse 2004, 4). Das bedeutet insbesondere, dass Kategorien und Strukturen in der Semantik (aber auch in der Syntax, Morphologie und Phonologie) nicht unabhängig existieren, sondern erst durch unsere kognitiven Fähigkeiten entstehen – und zwar auf Grundlage von konkreten Äußerungen und in spezifischen Situationen, also im Sprachgebrauch.

Die wesentlichen Merkmale der Kognitiven Linguistik holistischer Prägung sind damit die Ablehnung der Autonomiehypothese, eine Priorisierung sprachlicher Bedeutung, die sich aus der zentralen Rolle konzeptueller Strukturen ergibt, sowie die in der dritten These formulierte Gebrauchsorientierung, aufgrund derer häufig auch mit der Bezeichnung »usage-based« auf die entsprechenden Ansätze Bezug genommen wird. Mit der Abwendung von der Untersuchung der Kompetenz (die bei Annahme eines autonomen Sprachmoduls mit diesem identifiziert wird) hin zur Untersuchung der Performanz sieht sich die Kognitive

ein sprachlicher Ausdruck in einer konkreten sprachlichen Situation interpretiert wird (siehe dazu Langacker 2008, 55f.). Der Begriff des *construal* bzw. die mit ihm verbundenen Prozesse werden in Abschnitt 3.2.2 erneut aufgegriffen, im Rahmen von Cruses Konzeption der Bedeutungskonstitution, die er als *dynamic construal approach* bezeichnet.

25 In Bezug auf Systeme (hier: das Sprachsystem) bezeichnet Emergenz die Herausbildung von komplexen Eigenschaften auf Grundlage des Zusammenspiels der Einzelteile, wobei sich diese neuen Eigenschaften nicht auf die der Einzelteile zurückführen lassen (ganz im Sinne der Redensart »das Ganze ist mehr als die Summe seiner Teile«).

Linguistik in einer langen Tradition gebrauchsorientierter Theorien von Sprache, die von den oben genannten Arbeiten von Langacker über den von J. R. Firth vertretenen Kontextualismus[26] bis zurück zu frühen strukturalistisch geprägten Ansätzen wie dem Distributionalismus nach Zellig Harris reicht (siehe etwa Harris 1954; 1968).[27]

3.1.2 Sprache als semantisches Wissen

Die im vergangenen Abschnitt vorgenommene Eingrenzung auf den holistischen Ansatz der Kognitiven Linguistik ist im Kontext dieser Arbeit, insbesondere im Hinblick auf die computerlinguistische Modellierung, vor allem deshalb von Interesse, da dies direkte Auswirkungen darauf hat, wie die Repräsentation linguistischen Wissens gesehen wird. Nach Geeraerts (2006a, 3) ist das fundamentale Prinzip der Kognitiven Linguistik, »that language is all about meaning«, dass also Sprache als etwas primär Semantisches anzusehen ist und dass es somit auch bei der Untersuchung von Sprache zuallererst immer um Bedeutung geht. Damit bringt er zum Ausdruck, dass sich die Kognitive Linguistik nicht einfach mit der Untersuchung linguistischen Wissens beschäftigt (wie etwa der Generativismus), sondern dass sie Sprache selbst als eine Form von Wissen ansieht, welches nur mit dem Schwerpunkt auf der Semantik untersucht werden kann (siehe Geeraerts 2006a, 3). Dieses Grundprinzip wird von Geeraerts mittels vier ergänzender Grundsätze ausformuliert (siehe Geeraerts 2006a, 4–6), in denen er im Wesentlichen die im vergangenen Abschnitt beschriebenen Thesen auf sprachliche Bedeutung anwendet, um daraus den spezifischen Bedeutungsbegriff der Kognitiven Linguistik zu entwickeln:

- »Linguistic meaning is perspectival« (siehe Geeraerts 2006a, 4). Hierin spiegelt sich die Verneinung einer objektivistischen Sicht wider, insofern die Berücksichtigung der Perspektive impliziert, dass Bedeutung als sprechergebunden und damit als im konkreten Sprachgebrauch (der Performanz) verortet verstanden werden muss.

26 Firth' Kontextualismus steht für eine spezifische Ausprägung des Strukturalismus, die den Sprachgebrauch (die *parole*) zum zentralen Untersuchungsgegenstand machte (siehe u.a. Firth 1957). Aus Firth' Kontextualismus ging u.a. eine spezifisch englische Tradition korpusbasierter Ansätze hervor, von der auch Cruses Arbeiten beeinflusst sind (siehe u.a. Sinclair 1991; McEnery/Wilson 2001). Firth wird zudem auch in der Computerlinguistik, v.a. im Kontext probabilistischer Ansätze, oftmals als Referenz angegeben, so auch im Zusammenhang mit dem Word Space Model (siehe dazu Kapitel 4).

27 Nach McEnery/Wilson (2001) ist im Grunde die gesamte Linguistik vor Chomsky als korpusbasiert anzusehen; Harris selbst kommt bei ihnen nicht sonderlich gut weg: so bezeichnen sie seine Sicht, dass Sprache sich vollständig aus Korpora erschließen lasse, als »bullish« (siehe McEnery/Wilson 2001, 7). Sie erkennen jedoch an, dass viele seiner Ideen nach wie vor großen Einfluss haben. Tatsächlich erfährt Harris mit der Wiederentdeckung empirischer Ansätze ein spätes Revival, zum einen in der Kognitiven Linguistik (vgl. z.B. Croft/Cruse 2004), zum anderen auch – ebenso wie Firth – im Kontext der sogenannten Distributional Semantic Models (siehe dazu u.a. Sahlgren 2006; 2008).

- Der zweite Grundsatz besagt, dass sprachliche Bedeutung »dynamic and flexible« ist (siehe Geeraerts 2006a, 4), da sie nur so auf Veränderungen in der Welt reagieren und damit die jeweils spezifischen Erfahrungen wiedergeben kann.
- Im dritten Grundsatz (»Linguistic meaning is encyclopaedic and non-autonomous«, siehe Geeraerts 2006a, 4) spiegelt sich die im vergangenen Abschnitt mit Croft und Cruse formulierte Aufhebung jeglicher Trennung von linguistischem Wissen und anderen Formen konzeptuellen Gehalts wider. Die Konsequenz ist, dass Sprache nicht als ein autonomes Sprachmodul realisiert sein kann, sondern im Kontext der allgemeinen kognitiven Fähigkeiten zu verstehen ist.
- Der vierte Grundsatz kann schließlich als die Konsequenz der ersten drei gesehen werden: »Linguistic meaning is based on usage and experience« (siehe Geeraerts 2006a, 5). Betrachtet man sprachliche Bedeutungen als sprechergebunden, dynamisch und durch konzeptuelle Strukturen realisiert, so kommt den individuellen (sprachlichen wie nicht sprachlichen) Erfahrungen eine zentrale Rolle beim Aufbau des sprachlichen Wissens zu. Linguistisches Wissen ist demnach in der sprachlichen Erfahrung des Sprechers begründet und damit im konkreten Sprachgebrauch.

Gebrauchsorientierung und Konzeptualisierung gehören damit in der Kognitiven Linguistik zusammen: Nur über die Idee einer Emergenz von Struktur aus dem Sprachgebrauch lässt sich die Vorstellung aufrechterhalten, dass konzeptuelle Strukturen im Zuge der sprachlichen Erfahrungen aufgebaut werden bzw. sich verändern. Die Konsequenz für den Bedeutungsbegriff ist, dass die Variabilität von Bedeutung aus Sicht der Kognitiven Linguistik als eine essentielle Eigenschaft von sprachlichen Ausdrücken anzusehen ist, welche die Produktivität und Ausdrucksstärke von Sprache überhaupt erst ermöglicht.

3.2 Kognitive Semantik

Wie im vergangenen Abschnitt verdeutlicht ist das Leitmotiv der Kognitiven Linguistik die Auseinandersetzung mit sprachlicher Bedeutung. Diese wird als sprechergebunden sowie als hochgradig dynamisch und flexibel angesehen. Sprachliche Bedeutung ist eingebettet in andere Formen (linguistischen) Wissens und resultiert letztlich aus Sprachgebrauch und -erfahrung. Im Grunde ist somit die gesamte Kognitive Linguistik als ›semantikzentriert‹ anzusehen; dennoch wird die Auseinandersetzung mit sprachlicher Bedeutung unter der Bezeichnung »Kognitive Semantik« als eigenständiger Bereich innerhalb der Kognitiven Linguistik behandelt. Ein zentrales Anliegen der Kognitiven Semantik ist es, die Dynamik und Flexibilität sprachlicher Bedeutungen erklären zu können. Nachdem dies nach Glynn/Fischer (2010) in den frühen Arbeiten der Kognitiven Semantik oftmals

noch unter Rückgriff auf strukturalistisch geprägte Konzepte geschah,[28] wurde schon bald die Notwendigkeit einer Neuorientierung des Bedeutungsbegriffs deutlich, hier stellvertretend formuliert durch Geeraerts:

> The tremendous flexibility that we observe in lexical semantics suggests a procedural (or perhaps ›processual‹) rather than a reified conception of meaning; instead of meanings as things, meaning as a process of sense creation would seem to become our primary focus of attention. (Geeraerts 1993, 260)

In diesem Abschnitt soll ein solch ›prozessualer‹ Bedeutungsbegriff, wie ihn Geeraerts hier einfordert, entwickelt werden; dabei werden insbesondere auch die bereits in der Einleitung eingeführten Konzepte des Bedeutungspotentials sowie der Bedeutungskonstitution aus Sicht der Kognitiven Semantik präzisiert. Grundlage hierfür bildet vor allem Cruses *dynamic construal approach* (siehe Croft/Cruse 2004; Cruse 2004; 2010), in dem er die Grundannahmen der Kognitiven Linguistik bezüglich der Natur sprachlichen Wissens auf den Bereich sprachlicher Bedeutungen, insbesondere auf Wortbedeutung anwendet.

Ausgangspunkt für Cruse ist die sogenannte enzyklopädische Sicht auf Sprache, der zufolge Bedeutungen als konzeptuelle Strukturen angesehen werden und somit über ein rein sprachliches Wissen hinausgehen (Abschnitt 3.2.1). Daraus folgt, dass Wörter nicht einfach fest über eine (oder mehrere) Bedeutung(en) verfügen, sondern vielmehr über ein ›Potential zu bedeuten‹. Die konkreten Bedeutungen sind damit nicht als aufzählbare Einheiten im Lexikon anzusehen, vielmehr entstehen sie erst in der tatsächlichen Verwendung (Abschnitt 3.2.2). Abschließend wird auf Grundlage von Cruses *dynamic construal approach* ein einfaches Prozessmodell skizziert (Abschnitt 3.2.3), anhand dessen die wesentlichen Ziele der Modellierung abgesteckt werden können.

3.2.1 Bedeutung als Potential

Im Rahmen seines *dynamic construal approach* sieht es Cruse als wesentliche Anforderung bei der Beschreibung der Beziehung zwischen Wörtern und Bedeutungen, sowohl das Auftreten festgelegter struktureller Eigenschaften im Lexikon (zum Beispiel deren Morphologie) als auch die offenkundig unendliche Flexibilität von Bedeutungen im Kontext in einem gemeinsamen Erklärungsansatz zu vereinen (siehe dazu Croft/Cruse 2004, 97). In rationalistischen Ansätzen wie der Generativen Grammatik wurde dies in der Regel dadurch gelöst, dass die Strukturinformationen im Lexikon verortet wurden, so dass die Variabilität

28 So zeigt etwa Geeraerts (1993), dass sich Lakoff (1987) in seiner Analyse mehrdeutiger Ausdrücke mittels sogenannter *radial networks* noch immer auf die Annahme von (Teil-)Bedeutungen als diskrete Einheiten stützt – eine Position, die sich auch in früheren Arbeiten von Cruse findet (siehe z.B. Cruse 1986).

von Bedeutungen durch Regeln und Prinzipien der Pragmatik erklärt werden konnte.[29] Damit wurde der Bereich der Semantik gewissermaßen in das Lexikon ›ausgelagert‹, indem er als eine von der Syntax unabhängige Komponente einer ansonsten vor allem grammatisch orientierten Theorie angesehen wird: Mögliche Bedeutungen werden einfach im Lexikon verortet, die Grammatik wiederum stellt ›semantische‹ Regeln bereit, welche die Auswahl steuern.[30]

Die Kognitive Semantik stellt hierzu eine radikale Gegenposition dar, deren Grundgedanke darin besteht, dass sprachliche Bedeutungen eben gerade nicht im Lexikon verortet sind, sondern dass diese vielmehr als konzeptuelle Strukturen bzw. als »manifestation of conceptual structure« (siehe Evans/Green 2006, 156) anzusehen sind. Wie in der holistisch ausgerichteten Kognitiven Linguistik insgesamt wird nicht zwischen semantischer und konzeptueller Ebene unterschieden, vielmehr wird beides zusammen gedacht: Semantisches Wissen und Weltwissen sind miteinander eng verwoben. Die Tatsache, dass semantische und konzeptuelle Struktur als gleichartig angesehen werden können, heißt jedoch nicht, dass sie identisch sind: Nicht alle Konzepte haben eine sprachliche Entsprechung, die mit Wörtern assoziierten Bedeutungen bilden stattdessen nur eine Teilmenge der insgesamt möglichen Konzepte – semantische Strukturen sind damit als eine Teilmenge der konzeptuellen Strukturen anzusehen (siehe dazu Evans/Green 2006, 159), schematisch dargestellt in Abb. 3.1.

Wortbedeutungen werden in der Kognitiven Semantik damit als enzyklopädisches Wissen verstanden, das heißt als eingebunden in das allgemeine Weltwissen, welches selbst nicht zwingend rein sprachlich gefasst sein muss. Nach Evans/Green können Bedeutungen schon allein aufgrund ihrer konzeptuellen Natur nicht einfach als Liste von Lexikoneinträgen definiert sein:

> […] words do not represent neatly packaged bundles of meaning (the dictionary view), but serve as ›points of access‹ to vast repositories of knowledge relating to a particular concept or conceptual domain […]. (Evans/Green 2006, 160)

In dieser Sicht dienen Wörter als direkte Verweise (»points of access«) auf konzeptuellen Gehalt, aus dem sie ihre Bedeutungen beziehen.[31] Jedes Wort verweist potentiell auf eine Vielzahl an möglichen Bedeutungen, die sich in einer Vielzahl an möglichen Zuordnungen zu konzeptuellen Strukturen äußern. Mit anderen Worten verfügen sprachliche Ausdrücke über ein abstraktes »Bedeutungspotential«, das

29 Nicht zuletzt auch durch die Dominanz rationalistischer Ansätze, allen voran der Generativen Grammatiktheorien in Chomsky'scher Prägung, wurden lexikalische Bedeutungen bzw. das Lexikon selbst auch in der Computerlinguistik lange Zeit als eine weitgehend statische Sammlung von Elementen angesehen.

30 So verlangt bspw. die Government and Binding Theory (Chomsky 1981) die Erfüllung bestimmter semantischer Rollen, die im Lexikon über einen sogenannten »Subkategorisierungsrahmen« spezifiziert sind.

31 Mit der Idee, Wörter als »point of access« zu begreifen, beziehen sich Evans/Green (2006) explizit auf Langacker (siehe dazu Langacker 1987, 163).

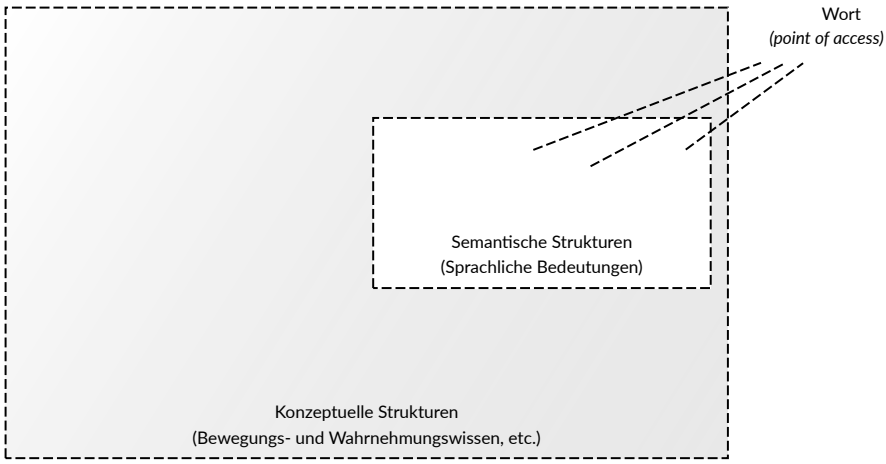


Abbildung 3.1: Die ›enzyklopädische Sicht‹ auf sprachliche Bedeutung. Wörter dienen als *points of access* zur konzeptuellen Ebene, wobei sie auf eine Vielzahl von möglichen Bedeutungen verweisen können. Diese sind dabei als Bestandteil des ›Weltwissens‹ anzusehen; semantische Strukturen bilden damit eine Teilmenge der konzeptuellen Strukturen.

die Menge der möglichen Konzepte umfasst, auf die der Ausdruck referieren kann. Das Bedeutungspotential geht dabei über eine reine Aufzählung möglicher Bedeutungen hinaus, da die damit verbundenen Festlegungen der Dynamik und Flexibilität sprachlicher Bedeutungen nicht gerecht werden; das Bedeutungspotential muss vielmehr als eine flexible Struktur verstanden werden – oder um es mit den Worten von Geeraerts zu formulieren:

> The dynamism of meaning does not just imply that it is easy to add new meanings to the semantic inventory of an expression, but also that we should not think of this overall structure of meanings as stable. (Geeraerts 2006a, 10)

Das Bedeutungspotential als flexible Struktur zu verstehen, bedeutet zunächst einmal nur, dass es nicht fixiert bzw. fixierbar ist. Das wiederum heißt nach Evans/Green (2006, 161f.) jedoch nicht, dass Wörter nicht dennoch mit bestimmten, konventionalisierten Bedeutungen (also einer Art ›Grundbedeutung‹) assoziiert sein können.[32] Analog zu den konzeptuellen Strukturen selbst bildet auch das Bedeutungspotential keine abgeschlossene Struktur, sondern befindet sich in einem

32 Dabei verfügen nicht alle Wörter über die gleichen Bedeutungsmöglichkeiten: Das Bedeutungspotential umfasst laut Evans/Green immer nur eine begrenzte »range of meanings« bzw. bildet gemäß der hier gewählten Terminologie immer nur einen bestimmten »point of access«, der auf eine bestimmte (enzyklopädische) ›Grundbedeutung‹, d. h. auf ein Konzept verweist. Diese Grundbedeutungen beruhen auf Konventionen, sie ergeben sich aus der ›Verwendungsgeschichte‹ der Wörter, im Zuge derer das Bedeutungspotential gewissermaßen ›erworben‹ wird (vgl. Evans/Green 2006, 161f.).

stetigen Wandel: So wie sich die konzeptuellen Strukturen durch Gebrauch und Erfahrung ändern, so ändern sich damit auch die möglichen Verweise auf diese Strukturen, die mit dem Bedeutungspotential verbunden sind. Grenzt man dies auf Wortbedeutungen ein, so heißt das, dass sich das Bedeutungspotential von Wörtern mit jeder Verwendung ändert und dass mit jeder Verwendung potentiell neue Bedeutungen hinzukommen können.

Im Rahmen seines *dynamic construal approach* nimmt Cruse eine zusätzliche Differenzierung des Bedeutungspotentials vor. So unterscheidet er einerseits zwischen *purport*, was sich mit ›konzeptuellem Gehalt‹ übersetzen lässt, und andererseits einem *set of conventionalized constraints*, die unmittelbar mit dem *purport* verbunden sind (siehe Abb. 3.2). Nach Cruse verfügt jedes Wort über einen Bedeutungsgehalt im Sinne des *purport*:

> Each lexical item (word form) is associated with a body of conceptual content that is here given the name purport. […] Purport may consist of a relatively coherent body of content, or it may display relatively disjunct parts (as in traditional ›homonymy‹); or, indeed any intermediate degree of coherence or lack of it. (Croft/Cruse 2004, 100)

Der konzeptuelle Gehalt im Sinne des *purport* ist dabei keinesfalls fix: »every experience of the use of a word modifies the word's purport to some degree.« (Croft/Cruse 2004, 101) Mehr noch, *purport* ist »essentially non-semantic« (103), also eine Art sprachliches ›Rohmaterial‹, das für sich genommen in dem Sinne als abstrakt anzusehen ist, dass es nicht weiter ausgedeutet ist. Ebenfalls Teil des

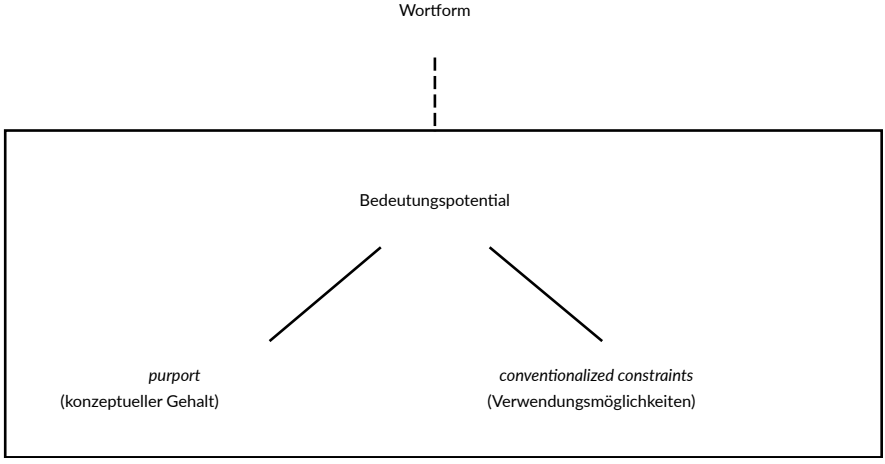


Abbildung 3.2: Differenzierung des Bedeutungspotentials in *purport* und *conventionalized constraints*. Der *purport* schränkt die Bedeutungsmöglichkeiten auf bestimmte Konzepte ein; die *constraints* wiederum bestimmen die Art und Weise, wie ein Wort verwendet werden kann.

Bedeutungspotentials sind die *conventionalized constraints*. Diese entstammen der Sprechergemeinschaft und spiegeln die Art und Weise wider, wie Wörter normalerweise (das heißt konventionell) verwendet werden.

Purport und *conventionalized constraints* sind dabei fest miteinander verbunden: Zum einen bestimmt der mit einer Wortform assoziierte *purport* das grundlegende semantische Potential in Abgrenzung zu anderen Wortformen. Zum anderen ist durch Konventionen gesteuert, wie das Wort verwendet werden kann, und damit auch, welche verschiedenen Ausdeutungen einer Wortform auf Grundlage des assoziierten *purport* überhaupt möglich sind. Weder *purport* noch *conventionalized constraints* sind selbst sprachlich explizierbar; somit ist auch das Bedeutungspotential insgesamt als eine abstrakte Struktur anzusehen. Die mit dem Begriff des Bedeutungspotentials verbundene Vorstellung von ›Bedeutungsmöglichkeiten‹ ist deshalb nicht einfach als eine Menge an möglichen Bedeutungen (im Sinne von Lesarten) zu verstehen, sondern als ›Möglichkeiten zu bedeuten‹, über die Wörter ganz grundsätzlich verfügen – wobei diese Möglichkeiten in ihrer konkreten Realisierung mittels Konventionen eingeschränkt sind.

3.2.2 Bedeutung als Prozess

Die radikale Konsequenz dieser Sichtweise ist, dass Wortbedeutungen im klassischen Sinne nicht existieren: Bedeutungen liegen in der konzeptuellen Struktur begründet, nicht im Wort bzw. sprachlichen Ausdruck – Bedeutungen sind demnach keine festen Entitäten im Lexikon, sondern flexible Verweise auf konzeptuellen Gehalt. Das lexikalische Wissen ist vielmehr reduziert auf die möglichen Bedeutungsweisen, mit denen Wörter assoziiert sind, das heißt auf ihr jeweiliges Bedeutungspotential, und erst im Zuge der Verwendung von Wörtern konstituieren sich konkrete Bedeutungen. Evans/Green fassen diese Konzeption von Bedeutung wie folgt zusammen:

> […] language itself does not encode meaning. Instead […] words (and other linguistic units) are only ›prompts‹ for the construction of meaning. […] meaning is constructed at the conceptual level: meaning construction is equated with **conceptualisation** […] It follows from this view that meaning is a process rather than a discrete ›thing‹ that can be ›packaged‹ by language. (Evans/Green 2006, 162, Hervorhebung gemäß Original)

Sprache vermittelt Bedeutung, aber diese ist nicht in der Sprache selbst enthalten. Wörter selbst ›haben‹ für sich genommen (also in isolierter Betrachtung) somit keine Bedeutung; stattdessen dienen sie mittels ihres Bedeutungspotentials als spezifische *points of access* zur eigentlichen Bedeutung, was hier heißt: zu konzeptuellem Gehalt. Vor diesem Hintergrund sind Bedeutungen nicht als Entitäten anzusehen, sondern vielmehr als ein *Prozess*. Wie bereits in der Einleitung angeführt,

lässt sich dieser Prozess in der Differenz von abstraktem Bedeutungspotential und konkreter Bedeutung verorten: Die Wörter[33] dienen zunächst nur als Ausgangspunkt (*prompts*) für den auf konzeptueller Ebene erfolgenden Prozess der Bedeutungskonstitution (bei Evans/Green: *meaning construction*), der als »›selection‹ of an appropriate interpretation against the context of the utterance« (siehe Evans/Green 2006, 161) verstanden werden kann, also als Auswahl einer kontextuell angemessenen Interpretation der Äußerung.

Die Konzeption einer Bedeutungskonstitution auf Grundlage eines abstrakten Bedeutungspotentials ist auch die Grundlage für Cruses *dynamic construal approach*, dem zufolge Bedeutungen *online*, also erst im Zuge konkreter Verwendungen entstehen. Wo Evans/Green von »meaning construction« sprechen und diese mit dem Prozess der Konzeptualisierung identifizieren, gibt Cruse dem von Langacker geprägten, weniger technisch als vielmehr psychologisch konnotierten Begriff des *construals* den Vorzug, welcher den Prozess der Deutung bzw. Interpretation eines sprachlichen Ausdrucks in einer konkreten sprachlichen Situation bezeichnet (siehe Langacker 2008, 55f.). Auch in Cruses Konzeption wird Bedeutung somit als ein Prozess verstanden: Wörter bringen mittels ihres Bedeutungspotentials zunächst nur den unausgedeuteten *purport* mit ein (quasi als semantisches Rohmaterial) sowie die mit dem *purport* verbundenen *conventionalized constraints*. Zusammen bilden diese die Grundlage für das *construal* der konkreten Bedeutung:

> On this view, words do not really have meanings, nor do sentences have meanings: meanings are something that we construe, using the properties of linguistic elements as partial clues, alongside non-linguistic knowledge, information available from context, knowledge and conjectures regarding the state of mind of hearers and so on. (Croft/Cruse 2004, 98)

Das Bedeutungspotential im Sinne des *purport* ist zunächst nur einer von mehreren »partial clues« für das Erfassen der konkreten Bedeutung. Gleiches gilt für die *conventionalized constraints*, die als Teil des Bedeutungspotentials ebenfalls zu den Eigenschaften der zu betrachtenden sprachlichen Einheiten zählen. Analog zu den *prompts* im Zitat von Evans/Green weiter oben, die als *points of access* zu konzeptuellen Strukturen verstanden werden können, ist das Bedeutungspotential bei Cruse nur der Ausgangspunkt für den Prozess der Bedeutungskonstitution, in den zusätzlich zu diesen »partial clues« auch alle weiteren im Kontext verfügbaren Informationen einbezogen werden. So wird im Zuge der konkreten Verwendung über den Kontext eine Reihe von zusätzlichen Bedingungen mit eingebracht, die Cruse unter der Bezeichnung *contextual constraints* zusammenfasst. Von besonderer Bedeutung, vor allem im Hinblick auf eine computerlinguistische Modellierung, sind dabei diejenigen *constraints*, die sich aus dem direkten

33 Bzw. mit Evans/Green: die mit ihnen assoziierten ›konventionalisierten Bedeutungen‹ bzw. ›Grundbedeutungen‹.

linguistischen Kontext ergeben, als die einzig ›sichtbaren‹ Bedingungen. Gemeint ist hier sowohl das unmittelbare linguistische Umfeld als auch der diskursive Kontext, einschließlich der Art des Diskurses bzw. des Texttyps, insofern dieser bestimmte Verwendungen begünstigt (zum Beispiel Zeitungstexte im Kontrast zu Chat-Kommunikation).[34]

Die eigentliche Bedeutungskonstitution beschreibt Cruse schließlich über den Begriff des *construals* als einen mehrstufigen Prozess, im Zuge dessen das Bedeutungspotential in vollständig kontextualisierte und damit konkrete Bedeutungen transformiert wird (siehe dazu Croft/Cruse 2004, 103f.), schematisch dargestellt in Abb. 3.3. In diesem Prozess kommen die im Kontext verfügbaren *contextual constraints* zum Tragen, indem sie im Zusammenspiel mit den *conventional constraints*, welche die Wortform vermittels ihres Bedeutungspotentials selbst mitbringt, das *construal* regulieren und auf diese Weise die Bedeutungsmöglichkeiten innerhalb konkreter Kontexte schrittweise einschränken. Daraus ergeben sich auf jeder Zwischenstufe sogenannte *pre-meanings*, welche als vorläufige Resultate von (elementaren) Teilprozessen des *construals* anzusehen sind (siehe Croft/Cruse 2004, 103f.) – die vollständig kontextualisierte und damit konkrete Bedeutung liegt in Cruses Konzeption hingegen erst nach Abschluss aller *construal*-Operationen vor.

Im Hinblick auf die computerlinguistische Modellierung spielt an dieser Stelle die Unterscheidung zwischen sogenannten *default construals* und den *full contextual construals* eine entscheidende Rolle.[35] Während das *default construal* auf den konventionellen *constraints* beruht, welche Teil des Bedeutungspotentials sind, kommen im Zuge des *full contextual construals* auch die im linguistischen Umfeld enthaltenen kontextuellen *constraints* zum Tragen. Die aus dem *default construal* resultierenden *pre-meanings* entsprechen dabei im Wesentlichen den weiter oben eingeführten konventionalisierten Bedeutungen und damit einer Art ›Grundbedeutung‹. Sind die *conventional constraints* sehr stark, dann widerstehen sie kontextuellen *constraints*, und die Grundbedeutungen werden als Ergebnis des gesamten *construal*-Prozesses interpretiert. In der Regel sind die *conventional constraints* jedoch relativ schwach, und die Grundbedeutungen werden durch

34 Die *contextual constraints* umfassen in Cruses Konzeption darüber hinaus auch den physischen und sozialen Kontext sowie das Wissen über bisher erfahrene *construals* – also im Grunde alles, was in der Kommunikation eine Rolle spielt. Schon hier wird deutlich, dass eine computerlinguistische Modellierung dies bestenfalls in Teilen umsetzen kann.

35 Während die detaillierte Differenzierung verschiedener *construal*-Operationen für die Theoriebildung in der Kognitiven Linguistik von großem Interesse ist, würde dies im Hinblick auf die in dieser Arbeit angestrebte Modellierung zu weit gehen. So unterscheiden Croft/Cruse (2004, 46f.) zwischen einer Vielzahl an einzelnen linguistischen *construal*-Operationen, die sie unter den vier Hauptkategorien »Attention/salience«, »Scope«, »Judgement/comparison« und »Constitution/Gestalt« zusammenfassen, welche jeweils »basic cognitive abilities«, also allgemeine kognitive Fähigkeiten bezeichnen. Die linguistischen *construal*-Prozesse sind selbst als Instanzen bzw. als »special cases of general cognitive processes described in psychology and phenomenology« anzusehen (Croft/Cruse 2004, 45) und umfassen u.a. verschiedene Formen der Kategorisierung und Perspektivierung.

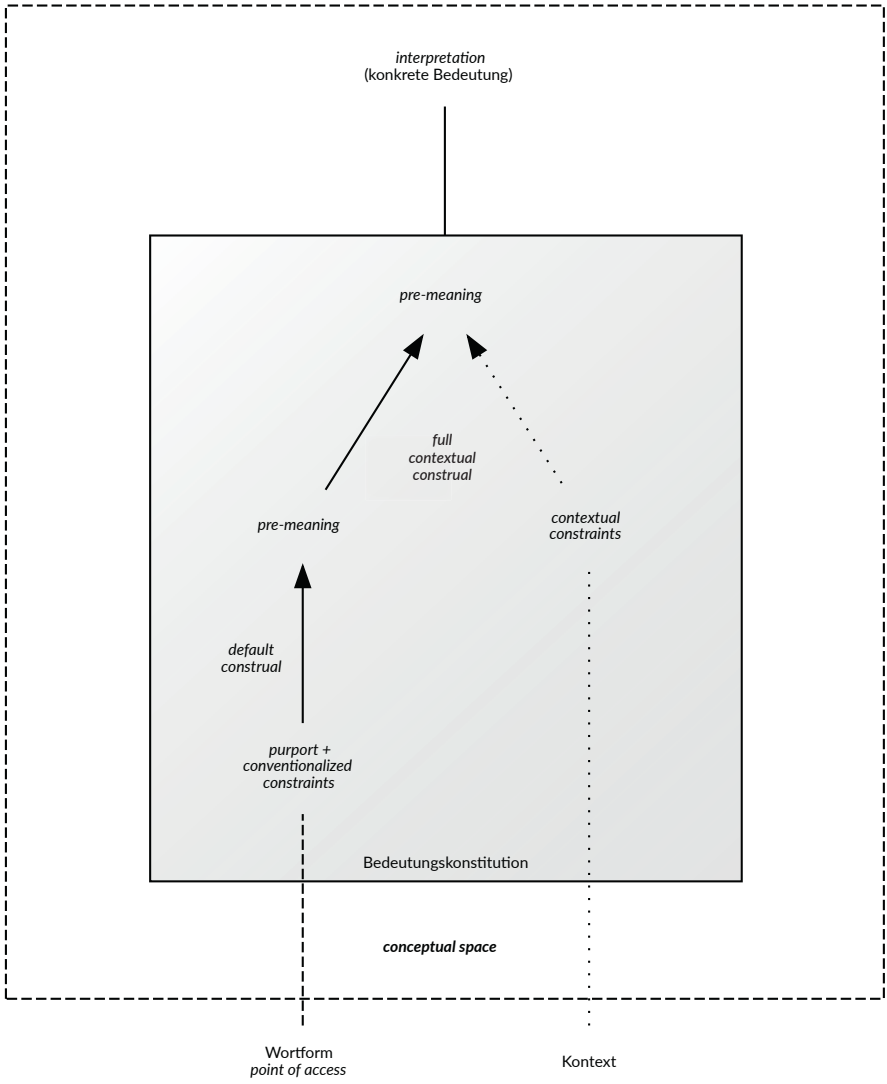


Abbildung 3.3: Prozess der Bedeutungskonstitution gemäß dem *dynamic construal approach* (siehe Croft/Cruse 2004). Die Wortform dient als *point of access* zur konzeptuellen Ebene und ist dort mit einem Bedeutungspotential assoziiert. Dieses wird im Zuge mehrstufiger *construal*-Operationen in eine konkrete Bedeutung überführt, gesteuert von den mit dem *purport* verbundenen *conventionalized constraints* sowie den im Kontext enthaltenen *contextual constraints*.

die mit den kontextuellen *constraints* verbundenen *construals* überschrieben. Die daraus resultierenden *fully construed meanings* entsprechen der konkreten, kontextualisierten Bedeutung des Wortes. Diese bezeichnet Cruse als *interpretations* (siehe Croft/Cruse 2004, 98), um zu verdeutlichen, dass es sich hierbei um flüchtige, zeit- und situationsgebundene Bedeutungen handelt.[36]

Mit der Unterscheidung zwischen *default construals* und *full contextual construals* lässt sich die hohe semantische Flexibilität sprachlicher Bedeutung erklären, ohne dabei die grundsätzliche Stabilität von Sprache vollständig in Frage zu stellen. Zwar verfügen Wörter über eine Art Grundbedeutung, grundsätzlich ist ihre Bedeutung jedoch nicht festgelegt, sondern ergibt sich erst aus dem Zusammenspiel mit ihren jeweiligen Kontexten. Die semantische Flexibilität von Sprache basiert zum einen auf der Beschaffenheit des Bedeutungspotentials, zum anderen auf der Sensitivität der *construal*-Prozesse gegenüber kontextuellen *constraints*. Die *conventionalized constraints* sorgen dafür, dass die kontextuelle Variablität sich in gewissen Grenzen abspielt. Indem sie ein *default construal* auslösen und dadurch immer zumindest eine *default interpretation* ermöglichen, haben die *conventionalized constraints* eine Art ›stabilisierende‹ Funktion für die Sprache insgesamt (siehe Croft/Cruse 2004, 103f.).[37]

3.2.3 Implikationen für die Modellierung

Nachdem mit der Beschreibung des *dynamic construal approach* nach Croft/Cruse (2004) die theoretischen Grundlagen der Bedeutungskonstitution erörtert wurden, steht im Folgenden die Frage im Mittelpunkt, wie sich diese theoretische Konzeption in ein computerlinguistisches Modell übertragen lässt, mit dem der Prozess der Bedeutungskonstitution simuliert werden kann. Der Fokus liegt dabei vor allem auf der Frage, was die Modellierung letztlich leisten muss, um – wie es in der Einleitung mit den Worten von Burghard Rieger formuliert wurde – Rückschlüsse auf den explikativen Wert der theoretischen Konzeption zu ermöglichen.

In der Drei-Ebenen-Unterscheidung nach David Marr (1982), an der sich die Gliederung dieser Arbeit orientiert (siehe Abschnitt 1.2), steht Cruses Konzeption somit für die erste Beschreibungsebene, auf der eine (abstrakte)

36 Cruses Begriff der *interpretation* meint somit nicht die Interpretation eines Wortes im Sinne einer Auslegung (wie etwa bei Evans/Green 2006), sondern das ›gedeutete Wort‹: »Interpretations are not contextual specifications of purports, they are transformations.« (siehe Croft/Cruse 2004, 101). Damit grenzt Cruse sich auch ganz explizit ab von seiner früheren Konzeption einer *contextual selection* (siehe dazu Cruse 1986), die sich anstelle einer Transformation noch auf die Annahme von (Teil-)Bedeutungen als diskrete Einheiten stützt.

37 Cruse sieht hierin einen möglichen Grund dafür, dass Bedeutungen oftmals als feststehend gesehen werden: so können die *default construals* mitunter den Eindruck einer vermeintlichen Eindeutigkeit von Bedeutungen erzeugen: »It is probably default construals that give the illusion of fixity of meaning.« Siehe Croft/Cruse (2004, 104).

Verarbeitungstheorie angegeben wird. Der Prozess der Bedeutungskonstitution wird hier mit Marr (1982) als ein informationsverarbeitender Prozess verstanden, bei dem im Wesentlichen eine Eingabeinformation in eine Ausgabeinformation überführt wird, schematisch dargestellt in Abb. 3.4.

Auf Grundlage dieses Schemas soll im Folgenden eine Abstraktion über Cruses Konzeption vorgenommen werden, indem zunächst die einzelnen Bestandteile (Input – Prozess – Output) charakterisiert werden. Cruses Konzeption zufolge lässt sich der Prozess der Bedeutungskonstitution als eine durch den Kontext motivierte Transformation eines abstrakten Bedeutungspotentials erfassen, im Zuge derer eine Konkretisierung der Bedeutung erfolgt. Die Eingabeinformation ist in diesem Falle somit das unkontextualisierte Wort bzw. das mit diesem assoziierte (rohe) Bedeutungspotential (bestehend aus *purport* und *conventionalized constraints*), sowie der konkrete Kontext, in dem das Wort auftritt und über den die sogenannten *contextual constraints* mit in den Prozess eingebracht werden. Die Ausgabeinformation bzw. das Ergebnis des Prozesses ist die konkrete Bedeutung des eingegebenen Wortes in genau diesem Kontext. Abb. 3.5 zeigt das entsprechend angepasste Schema der Bedeutungskonstitution.

Wesentliches Ziel der computerlinguistischen Modellierung wird es sein, eine algorithmische Entsprechung für diesen Prozess zu finden, die eine Simulation der Bedeutungskonstitution ermöglicht. Auf Grundlage der Simulation soll dann überprüft werden, ob das erwartete Ergebnis eintritt. Das Modell sollte demnach in der Lage sein, die Vorhersagen, die sich auf Grundlage der theoretischen Konzeption treffen lassen, einzulösen. Was also sind die Vorhersagen? Nach Cruse ist

Abbildung 3.4: Schematische Darstellung der Überführung einer Eingabeinformation (Input) in eine Ausgabeinformation (Output) durch einen informationsverarbeitenden Prozess.

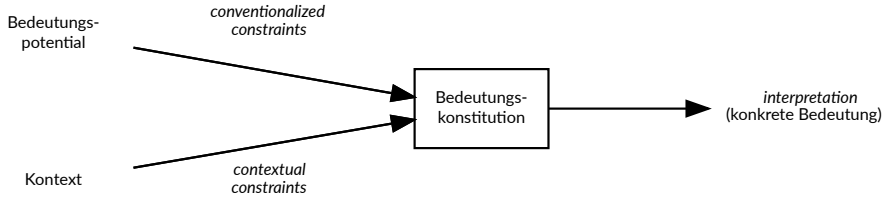


Abbildung 3.5: Vereinfachte Darstellung der Bedeutungskonstitution nach Cruse: Eingabe ist ein (abstraktes) Bedeutungspotential zusammen mit seinem lokalen Kontext; dieses wird im Zuge des Prozesses der Bedeutungskonstitution in eine konkrete Bedeutung transformiert.

die Bedeutungskonstitution ein grundlegender Prozess, der in jeder Verwendung zur Anwendung kommt, wobei potentiell aus jeder Kontextualisierung eine andere Bedeutung resultieren kann. Die Vorhersage der Theorie ist damit – stark vereinfacht – schlicht eine Variation der Bedeutung in verschiedenen Kontexten. Dabei ist zu beachten, dass diese Variation nach Cruse in den meisten Fällen nur minimal ausfällt, so dass mitunter eine »illusion of fixity of meaning« entsteht (siehe Croft/Cruse 2004, 104). Dies lässt sich anhand der folgenden Beispiele illustrieren:

Beispiel 3.1
a. Sie scheint gerne zu spielen.
b. Sie spielt eben einfach gerne.
c. Für sie scheint das keine Rolle zu spielen.
d. Sie scheint gerne Klavier zu spielen.

Die Vorhersage des Modells ist hier eine unterschiedliche Bedeutung für das Wort *spielen* in den verschiedenen Kontexten. Die Bedeutungsvariation in Beispiel 3.1 lässt sich in Cruses Konzeption dadurch erklären, dass das Bedeutungspotential unter dem Einfluss verschieden starker *contextual constraints* unterschiedlich stark transformiert wird. Während in den Kontexten 3.1.a und 3.1.b die konventionalisierte Bedeutung von *spielen* zum Tragen kommt, weicht die Bedeutung in den Kontexten 3.1.c und 3.1.d deutlich ab (»Rolle spielen« bzw. »Klavier spielen«). Dies lässt sich in Cruses Konzeption durch stärkere *contextual constraints* erklären. In

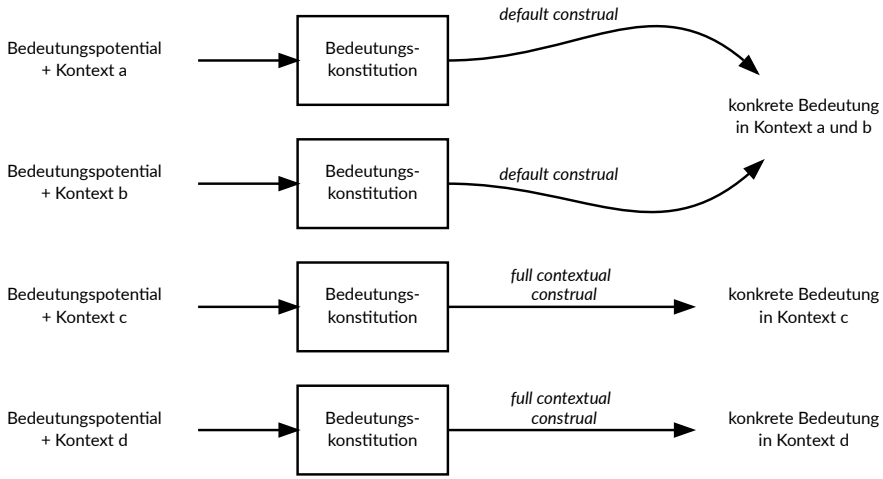


Abbildung 3.6: Variation der Bedeutung von *spielen* innerhalb der in Beispiel 3.1 aufgeführten Kontexte. Sind die *contextual constraints* sehr stark, so überschreiben sie das *default construal* (Kontext a und b), andernfalls wird zumindest das *default construal* vollzogen und eine Grundbedeutung kommt zum Tragen.

Abb. 3.6 wird dies dadurch dargestellt, dass aus verschiedenen Eingaben zum Teil eine abweichende, mitunter aber auch eine gleiche konkrete Bedeutung resultieren kann.

Die Überprüfung des Modells wird somit dann möglich, wenn die Vorhersage einer Variation auch im Modell sichtbar gemacht werden kann, das heißt, wenn es dort ein sichtbares Pendant zu diesem Effekt gibt. Ein computerlinguistisches Modell der Bedeutungskonstitution muss also in der Lage sein, zu zeigen, dass sich die konkrete Bedeutung – hier als das Ergebnis des Prozesses – in jedem Kontext zumindest leicht ändert. Ziel der Modellierung ist es somit, genau dies zu zeigen: Wenn es möglich ist, dass sich eine allgemeine Bedeutungsvariation in Abhängigkeit vom Kontext ablesen lässt, dann spricht im Grunde nichts dagegen, aus dieser (sichtbaren) Variation darauf zu schließen, dass sich die Bedeutung immer erst im lokalen Kontext konstituiert.

3.3 Zusammenfassung

Bevor im Folgenden die Konzeption einer Bedeutungskonstitution noch einmal im Hinblick auf die Modellierung zusammengefasst wird, wird an dieser Stelle zunächst noch eine Einordnung vorgenommen. Insbesondere muss betont werden, dass es sich bei der Kognitiven Semantik wie auch bei der Kognitiven Linguistik insgesamt nicht etwa um einen in sich geschlossenen Theorierahmen handelt. Wie bereits in der Einleitung zu diesem Kapitel angemerkt, ist die Kognitive Linguistik keine spezifische Theorie (siehe Evans/Green 2006, 3) und auch »kein einheitlich definierter Forschungsbereich« (siehe Schwarz 2008, 41), »[e]ine verbindliche und einheitliche Definition oder Eingrenzung des Bereichs Kognitive Linguistik gibt es jedenfalls derzeit nicht« (41). Nach Geerarts (2006) ist die Kognitive Linguistik stattdessen als ein flexibler Bezugsrahmen anzusehen, der als relativ junge Forschungsrichtung in seiner spezifischen Ausprägung noch weitgehend offen ist:

> Cognitive Linguistics is a flexible framework rather than a single theory of language […] it constitutes a cluster of many partially overlapping approaches rather than a single well-defined theory that identifies in an all-or-none-fashion whether something belongs to Cognitive Linguistics or not. (Geerarts 2006, 2)

Dementsprechend ist auch der *dynamic construal approach* nur als ein möglicher Ansatz unter vielen anzusehen. Tatsächlich ist die Idee einer Bedeutungskonstitution, bezogen auf ein flexibles Bedeutungspotential, nicht neu. So beschreibt zum Beispiel schon Rieger (1977, 59f.) – unter Bezugnahme auf Lyons (1971) – Bedeutung als einen Prozess zunehmender Einschränkung von Wahlmöglichkeiten. Mit seiner Konzeption einer prozeduralen Semantik (vergleiche unter anderem Rieger 1985) nimmt er zudem einige der in Croft/Cruse (2004) formulierten Ideen vorweg,

insbesondere die Prozesshaftigkeit von Bedeutungen sowie die Verlagerung des Untersuchungsgegenstands von der Kompetenz hin zur Performanz.

Auch die Annahme einer Nichtexistenz sprachlicher Bedeutung wurde nicht erst durch Cruse eingeführt. Nach Cruse wurde dieser Grundgedanke zuerst durch Moore/Carling (1982) formuliert (in ihrem Buch mit dem programmatischen Titel »Understanding language: towards a post-Chomskyan linguistics«) und findet sich seither in verschiedener Ausprägung in einer Vielzahl von Ansätzen innerhalb der Kognitiven Semantik wieder. Dabei wird der Begriff der Bedeutungskonstitution – anders als in dieser Arbeit – zumeist nicht auf den Bereich der Wortbedeutung beschränkt, sondern bezieht sich häufig auf die Konstitution von Bedeutung im Sinne einer übergeordneten Satzbedeutung (vergleiche dazu etwa Schwarz 2008, 59f. und 189f. sowie Evans/Green 2006, 365f.).

Gemeinsam ist den verschiedenen Ansätzen im Wesentlichen die theoretische Konzeption einer Bedeutungskonstitution als Differenz zwischen einem allgemeinen, zunächst unspezifischen Bedeutungspotential und einer konkreten Bedeutung. In der konkreten Ausprägung weisen sie zum Teil jedoch deutliche Unterschiede auf, zum einen hinsichtlich der Art und Weise, wie das Bedeutungspotential repräsentiert wird, zum anderen in Bezug auf die Beschaffenheit des Kontextes, der berücksichtigt wird. Von diesen Faktoren hängt in hohem Maße ab, wie der Prozess der Bedeutungskonstitution letztlich operationalisiert werden kann – eine verbindliche Vorgabe für eine Operationalisierung gibt es seitens der Kognitiven Semantik jedenfalls bislang nicht.

Für die Operationalisierung im Rahmen dieser Arbeit dient im Wesentlichen Cruses *dynamic construal approach* als Leitbild, der im Hinblick auf die Modellierung wie folgt zusammengefasst werden kann: In der Kognitiven Semantik haben Wörter selbst keine Bedeutung, sondern verfügen vielmehr über ein flexibles Bedeutungspotential, das sich als abstrakte Verweisstruktur auf ›konzeptuelle Kategorien‹ begreifen lässt. Das Bedeutungspotential besteht in Cruses Konzeption aus einem abstrakten, unsemantischen ›Bedeutungsgehalt‹ (dem *purport*) sowie einer Reihe an konventionalisierten Bedingungen (*conventional constraints*); hinzu kommen im Zuge konkreter Verwendungen kontextuelle Bedingungen (*contextual constraints*). Auf Grundlage dieser Informationen wird ein sprachlicher Ausdruck im Kontext konkreter Verwendungen als ein bestimmtes Konzept gedeutet (*construed*). Die aus diesem Prozess resultierenden »contextually construed meanings« sind nach Cruse jedoch nicht einfach mit Konzepten gleichzusetzen. Erstere (also die konkreten Bedeutungen) bezeichnet Cruse vielmehr als *interpretations*, bei denen es sich um flüchtige, zeit- und situationsgebundene Bedeutungen handelt. Diese *interpretations* bzw. *contextualized interpretations* sind damit nur temporäre Verweise auf Konzepte bzw. auf konzeptuelle Strukturen und ausschließlich von ›lokaler‹ Gültigkeit. Damit lässt sich die Vorstellung einer Nichtexistenz von Bedeutungen nochmals präzisieren: Bedeutung existiert nur im Sinne von kontextualisierter Bedeutung. Zwar verfügen unkontextualisierte Wörter über ein ›semantisches Potential‹; doch erst die tatsächliche Verwendung

›haucht ihnen Leben ein‹ (siehe Croft/Cruse 2004, 98). Bedeutung ›entsteht‹ erst in der konkreten Verwendung, sie ›konstituiert‹ sich immer nur in konkreten Kontexten. Vor dem Hintergrund dieser Konzeption erweitert Cruse nochmals die Perspektive auf das Bedeutungspotential:

> We can portray the total meaning potential of a word as a region in conceptual space, and each individual interpretation as a point therein. Understood in this way, the meaning potential of a word is typically not a uniform continuum: the interpretations tend to cluster in groups showing different degrees of salience and cohesiveness, and between the groups there are relatively sparsely inhabited regions. (Croft/Cruse 2004, 109)

Bei Cruse umfasst das Bedeutungspotential somit letztlich nicht nur den »purport« und ein Set von konventionellen und kontextuellen »constraints«, sondern impliziert auch bereits die Gesamtheit der möglichen »interpretations«, die im Zuge der Bedeutungskonstitution entstehen können. Das Bedeutungspotential (hier: »total meaning potential«) ist als solches nicht explizierbar, es bildet keine abgeschlossene Struktur, vielmehr einen Bereich im »conceptual space«.

Aufgabe der Modellierung wird es sein, dies umzusetzen: Eine Repräsentation des »conceptual space«, in dem im Zuge von Kontextualisierungen Festlegungen auf konkrete Bedeutungen gemacht werden. Wie bereits oben festgestellt, geben Croft/Cruse (2004) hier selbst keine klare Operationalisierung vor. Nimmt man das Zitat jedoch wörtlich, so bietet sich unmittelbar das Word Space Model (WSM) nach Schütze (1992; 1993) an, um damit diesen »conceptual space« zu modellieren und die »interpretations« darin zu verorten. Weil es sich beim WSM um ein eigenständiges Modell für die Ermittlung und Darstellung von Bedeutungen handelt, das zudem üblicherweise mit einem rein distributionellen Bedeutungsbegriff assoziiert ist, der von dem hier dargestellten abweicht, wird das WSM in Kapitel 4 zunächst unabhängig von der eigenen Modellierung vorgestellt. Anschließend kann im Sinne der Beschreibungsebenen von Marr (1982) dargestellt werden, wie das WSM für die Modellierung der in diesem Kapitel vorgestellten theoretischen Konzeption – bzw. nach Marr: der Verarbeitungstheorie – eingesetzt werden kann.

4. Das Word Space Model

Im vergangenen Kapitel wurde die theoretische Konzeption einer Bedeutungskonstitution als Differenz zwischen einem allgemeinen Bedeutungspotential und einer aktuellen Bedeutung innerhalb konkreter Verwendungen eingeführt. Dieser Konzeption zufolge konstituiert sich Bedeutung immer nur in konkreten Kontexten, in denen jeweils nur ›Teile‹ des Bedeutungspotentials zum Tragen kommen. Um dies in ein computerlinguistisches Modell zu überführen, bedarf es zunächst einer angemessenen Repräsentation des Bedeutungspotentials sowie der Kontextinformationen: Die eben genannten Teile des Bedeutungspotentials müssen in der Repräsentation enthalten sein bzw. aus ihr abgeleitet werden können.

Da die Kognitive Linguistik hier selbst keine einheitliche Operationalisierung vorgibt (siehe dazu auch Abschnitt 3.3), soll die Modellierung im Rahmen dieser Arbeit auf Grundlage des Word Space Model (WSM) erfolgen. Denn obwohl es bereits eine Reihe von Arbeiten aus dem Bereich der Kognitiven Linguistik gibt, in denen das Modell erfolgreich im Zusammenspiel mit kognitiv orientierten Ansätzen eingesetzt wird,[38] ist das WSM in der Kognitiven Linguistik keinesfalls etabliert – im Gegenteil, es wird zum Teil sogar als konkurrierender oder gegensätzlicher Ansatz gesehen (siehe dazu zum Beispiel Lenci 2008). Tatsächlich weichen die Grundannahmen über die Natur sprachlicher Bedeutung in einigen Punkten voneinander ab; das WSM stellt jedoch im Gegensatz zur Kognitiven Linguistik eine elaborierte Methode zur korpusbasierten quantitativen Analyse von sprachlichen Einheiten bereit. Eines der Ziele der Arbeit ist deshalb, die Eignung des Modells als eine methodische Ergänzung zur Kognitiven Linguistik auszuloten: Um das von ihr ausgerufene, stark empirisch ausgerichtete Forschungsprogramm durchzuführen, erscheint die Öffnung hin zu quantitativen Ansätzen und computerlinguistischen Modellen im Grunde unumgänglich (siehe dazu zum Beispiel Glynn/Fischer 2010), um dadurch eine empirische Überprüfung und Falsifikation der theoretischen Annahmen zu ermöglichen.[39]

In Abschnitt 4.1 wird zunächst die Funktionsweise des Modells sowie seiner Varianten erläutert, um damit die Bausteine für die Modellierung zusammenzustellen. Bevor diese für die Modellierung eingesetzt werden können, werden in Abschnitt 4.2 zunächst noch die Unterschiede in den theoretischen Vorannahmen markiert und dabei erörtert, wie das Modell dennoch für die Modellierung eingesetzt werden kann. In Abschnitt 4.3 werden schließlich nochmals die Vorteile, aber auch die Probleme und Grenzen des Modells in Bezug auf die Modellierung eines kognitiv orientierten, dynamischen Bedeutungsbegriffs zusammengefasst.

38 Zu nennen sind hier neben anderen Pustejovsky/Jezek (2008), Peirsman u.a. (2008; 2010), Heylen u.a. (2015) sowie Heylen u.a. (2008).

39 Ohne diese müsste die Kognitive Semantik, wie in der Einleitung angemerkt, streng genommen als »Ideologie oder Spekulation« (siehe Rickheit u.a. 2010, 14) angesehen werden.

4.1 Grundkonzeption des Modells

Im Folgenden wird zunächst die Funktionsweise des Word Space Model (WSM) aus technischer Sicht vorgestellt. Dafür wird der Aufbau des Modells aus dem zugrundeliegenden Vector Space Model hergeleitet (Abschnitt 4.1.1). Im WSM können auf Grundlage einer Analyse der sprachlichen Umgebungen Vektoren erstellt werden, die Wörter anhand ihrer Verwendungsmuster räumlich repräsentieren. Anschließend werden verschiedene Typen von Wortvektoren unterschieden, die sich aus der jeweiligen Parametrisierung des WSM ergeben (Abschnitt 4.1.2). Neben der Erstellung von Wortvektoren ermöglicht das WSM auch die Repräsentation von Einzelvorkommen. Diese basieren auf dem Konzept der Kookkurrenz zweiter Ordnung, das in Abschnitt 4.1.3 erläutert wird, bevor in Abschnitt 4.1.4 nochmals eine kurze Zusammenfassung der technischen Aspekte des Modells erfolgt.

4.1.1 Der Wortraum

In seiner Grundkonzeption baut das WSM unmittelbar auf dem Vektorraummodell (Vector Space Model, VSM) auf, das seinen Ursprung im Information Retrieval hat (siehe Salton u.a. 1975; Salton/McGill 1983). Seinen Namen verdankt das VSM der zugrunde gelegten Metapher einer räumlichen Darstellung von Ähnlichkeiten, welche nach Manning/Schütze (1999, 539) neben der konzeptionellen Einfachheit einer der Gründe für die weite Verbreitung des Modells ist. Im VSM werden Dokumente als Merkmalsvektoren in hochdimensionalen Räumen dargestellt. Als Merkmale dienen dabei die in den Dokumenten enthaltenen Wörter (bzw. in der Terminologie des Information Retrieval die Terme). Im Vektorraum können Dokumentvergleiche als Vektorvergleiche umgesetzt werden, die beispielsweise als Grundlage für das Scoring, die Klassifikation oder das Clustering von Dokumenten eingesetzt werden können. Für den Vergleich werden die als Vektoren repräsentierten Dokumente anhand ihrer Richtung im Vektorraum zueinander in Beziehung gesetzt, schematisch dargestellt in Abb. 4.1. In diesem stark stilisierten Vektorraum sind die Vektoren V_2 und V_3 ähnlicher zueinander als zu Vektor V_1.

Der im Information Retrieval eingesetzte Vektorraum, in dem die Dokumente repräsentiert werden, wird durch die in den Dokumenten auftretenden Terme definiert, weshalb er oftmals als »term space« bezeichnet wird. Dieser wird durch eine Term-Dokument-Matrix definiert, in der für jedes Dokument die enthaltenen Terme mit ihren Häufigkeiten eingetragen sind (siehe Abb. 4.2).[40] Die Größe der Matrix – und damit auch die Dimensionalität des durch sie beschriebenen Vektorraums – richtet sich dabei nach der Größe des in den Dokumenten verwendeten Vokabulars.

40 Im Information Retrieval ist es zudem üblich, die Termhäufigkeiten zusätzlich zu gewichten; sehr verbreitet ist hier z.B. das sogenannte »tf.idf-Maß« (siehe dazu Anm. 47 in Abschnitt 4.1.2).

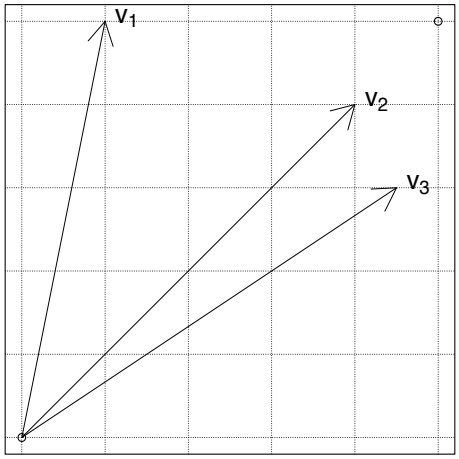


Abbildung 4.1: Schematischer Vektorraum mit den drei Vektoren V_1, V_2 und V_3.

Anhand dieser Matrix lässt sich gut die Übertragung auf Wortebene veranschaulichen: So wie Dokumente durch die in ihnen enthaltenen Terme beschrieben werden können, lassen sich in umgekehrter Perspektive auch die Terme bzw. Wörter durch die Dokumente beschreiben, die sie enthalten. Der Raum, in dem die Terme als Vektoren repräsentiert werden, wird in dieser Sicht durch die Dokumente definiert, so dass man ihn als *document space* bezeichnen kann. Die Dokumente stehen hierbei für den Kontext, in dem die Wörter auftreten. Ein entsprechendes Vorgehen zur Erstellung von Wortvektoren findet sich beispielsweise bei Salton (1971), die bekannteste Umsetzung ist jedoch das Modell der Latent Semantic Analysis (Deerwester u.a. 1990).

Die Beschaffenheit des Vektorraumes hängt unmittelbar von der Definition des Kontextes ab, innerhalb dessen das Auftreten von Wortformen bewertet wird. Nimmt man anstelle ganzer Dokumente einen kleineren Kontext, etwa nur Teile eines Dokuments oder nur das direkte Wortumfeld, so können Wörter durch die im entsprechenden Kontext auftretenden Wörter beschrieben werden. Während

	d_1	d_2	d_3	⋯	d_n
t_1	0	1	2	⋯	3
t_2	1	0	1	⋯	2
⋮	⋮	⋮	⋮	⋱	⋮
t_n	3	2	1	⋯	0

Abbildung 4.2: Beispiel für eine Term-Dokument-Matrix: Für jedes Dokument ($d_1 \ldots d_n$) wird in den Zeilen ($t_1 \ldots t_n$) die Häufigkeit der enthaltenen Terme eingetragen.

	t_1	t_2	t_3	⋯	t_i
t_1	0	1	2	⋯	3
t_2	1	0	1	⋯	2
⋮	⋮	⋮	⋮	⋱	⋮
t_j	3	2	1	⋯	0

Abbildung 4.3: Beispiel für eine Kookkurrenz-Matrix: Für jedes Paar von Termen (t_{ij}) wird die Häufigkeit des gemeinsamen Auftretens notiert. Die Größe der Matrix richtet sich dabei nach dem Vokabular des zugrunde gelegten Textkorpus.

im *document space* einfach das alleinige Vorkommen einzelner Wortformen in einer vordefinierten Einheit berücksichtigt wird, wird nun das gemeinsame Auftreten von Wörtern betrachtet, bezeichnet als »Kookkurrenz«. Für diese Konzeption eines Vektorraums auf Grundlage von Wörtern und ihrer Kookkurrenzen prägte Schütze (1992) den Begriff des *word space*. Im Wortraum wird jedes Wort durch die innerhalb einer vorgegebenen Kontextbreite auftretenden Elemente beschrieben. Daraus ergibt sich eine Kookkurrenzmatrix, in der die Häufigkeiten des gemeinsamen Vorkommens von Wörtern bzw. Wortformen festgehalten werden, schematisch dargestellt in Abb. 4.3.

Die Grundidee des WSM besteht nun darin, dass die durch die Zeilen und Spalten beschriebenen Vektoren das sprachliche Verhalten der jeweiligen Terme widerspiegeln, wie das folgende Zitat aus Schütze (1992, 2) verdeutlicht: »The approach here is to represent words as term vectors that reflect their pattern of usage in a large text corpus.« Da die Häufigkeiten des gemeinsamen Vorkommens in der Regel stark variieren, ergibt sich im Wortraum für jeden Term ein spezifisches Kookkurrenzmuster, das als Verwendungsmuster des jeweiligen Terms bzw. Wortes verstanden werden kann.

4.1.2 Wörter als Vektoren

In der Umsetzung des Wortraummodells muss nicht zwingend eine Matrix über alle Wörter erstellt werden, das heißt, dass die Mengen der beschreibenden und der beschriebenen Elemente nicht zwingend übereinstimmen müssen. Die Dimensionierung der zugrunde gelegten Matrix ist somit weitgehend variabel, sowohl hinsichtlich der Wörter, die durch Vektoren repräsentiert werden sollen (Vektoren können demnach auch einzeln und ad hoc erstellt werden), als auch in Bezug auf die Wörter, die als Merkmale der Kookkurrenzvektoren dienen sollen (und die Dimensionalität des Wortraums bestimmen). Ein entsprechendes Vorgehen zur Erstellung von Wortvektoren soll im Folgenden am Beispiel von Levy/Bullinaria (2001) verdeutlicht werden. Hier wird eine Kookkurrenzmatrix zugrunde gelegt, bei der nur die Spalten als Wortvektoren angesehen werden. Damit übernehmen die Wörter hier zwei unterschiedliche Funktionen: In den

The lorry *driver* swerved on the road. As well as causing *pollution*, a lorry also has large *wheels*. A lorry requires *diesel* to work. A lorry might carry *sweet* apples and bananas. Bananas are easier to *peel* than apples but apples have nicer *trees*. Bananas are cheaper than apples in a *shop*.

	lorry	apples	bananas
sweet	1	1	2
trees	0	2	2
shop	0	0	1
eat	0	0	0
peel	0	2	2
driver	1	0	0
road	1	0	0
diesel	2	0	0
pollution	1	0	0
wheels	2	0	0

Abbildung 4.4: Erstellung von Wortvektoren für einen Beispieltext. Die unterstrichenen Wörter werden als Zielwort, die kursiv gesetzten als Kontextwörter angesehen. Berücksichtigt wird die Kookkurrenz innerhalb einer Fensterbreite von fünf Elementen. Abbildung aus Levy/Bullinaria (2001, 3).

Spalten sind sie Zielwort der Betrachtung, in den Zeilen fungieren sie als Merkmale, deren Kookkurrenz zu den Zielwörtern in der jeweiligen Spalte notiert wird. Abb. 4.4 illustriert das Vorgehen anhand eines kurzen Beispieltextes.

Die Art des Vektorraumes – und damit auch die Beschaffenheit der darin repräsentierten Wortvektoren – ist eng verknüpft mit dem Begriff des Kontextes. Dieser wird einerseits von der Kontextbreite (im Beispiel fünf), andererseits von der Auswahl der Merkmale bestimmt (im Beispiel kursiv gesetzt). Zum einen werden in dem Beispiel nur für einige wenige Wörter Vektoren erstellt, zum anderen werden nicht alle Wörter als Kontextwörter betrachtet; so wird zum Beispiel das gemeinsame Autreten von *the* und *lorry* ebenso wenig gezählt wie das der Wörter *cheaper* und *apples*. Diese beiden Faktoren, das Kontextfenster und die Merkmalsauswahl, bestimmen maßgeblich, welche kontextuellen Informationen verwendet werden, wodurch verschiedenartige Kookkurrenzbeziehungen betont werden können.

Kontextfenster

Die Kontextbreite ist grundsätzlich variabel; dies können die direkten Nachbarn sein oder aber 10, 20, oder gar 100 Wörter. Nachbarschaft kann auch linguistisch definiert werden, etwa als gemeinsames Auftreten innerhalb einer linguistischen Einheit wie Phrase, Teilsatz oder Satz (bis hin zu Absätzen oder ganzen Dokumenten, vergleiche dazu Abschnitt 4.1.1). Werden bei einem breiteren Fenster eher thematische Relationen abgebildet, im Sinne einer Verwendung in einem gleichen oder ähnlichen Themenbereich (vergleichbar dem *document space*), liegt der Fokus bei einem engen Kontextfenster auf den lokalen Beziehungen, die ein

sprachlicher Ausdruck mit seinem direkten Umfeld unterhält. Dadurch werden in höherem Maße auch grammatische Relationen berücksichtigt. Zusätzlich kann zwischen rechtem und linkem Kontext unterschieden werden.[41]

Merkmalsauswahl
Neben der gewählten Kontextbreite spielt vor allem die Auswahl der Elemente eine Rolle, deren Kookkurenz innerhalb des gegebenen Fensters gemessen wird. Für die Auswahl der Merkmale gibt es eine Vielzahl verschiedener Strategien, deren Gemeinsamkeit in der Einsicht besteht, dass nicht alle Elemente des Vokabulars in gleichem Maße nützlich sind für die Beschreibung.[42] Die Merkmalsauswahl beginnt schon bei der Vorverarbeitung, wenn bspw. über die einfache Tokenisierung hinaus auch ein Stemming oder eine Lemmatisierung durchgeführt wird (was die Anzahl der möglichen Merkmale deutlich verringert), oder aber mittels Part-of-Speech-Tagging die Wortarten ermittelt werden, etwa um Verben nur durch ihre nominalen Komplemente oder Nomen nur durch die kookkurrierenden Adjektive zu beschreiben. Zusätzlich können auch einfache textstatistische Maße für die Filterung des Merkmalssets herangezogen werden. Eine sehr pragmatische Variante dieses Vorgehens verfolgen Levy/Bullinaria (2001), die einfach eine begrenzte Zahl der häufigsten Wörter eines Korpus als Merkmale einsetzen. Die Filterung auf Basis von textstatistischen Kriterien wird von Sahlgren als arbiträr bezeichnet (siehe Sahlgren 2006, 39), da sie in hohem Maße von der Beschaffenheit des zugrunde gelegten Korpus abhängt.[43]

Dimensionsreduktion
Alternativen zur einfachen Filterung des Merkmalssets finden sich unter anderem im Modell des Hyperspace Analogue to Language (HAL) (Lund/Burgess 1996; Burgess u.a. 1998; Burgess 1998), bei der Latent Semantic Analysis (LSA) (Dumais u.a. 1988; Deerwester u.a. 1990) sowie beim Random Indexing (RI) (Karlgren/Sahlgren 2001; Sahlgren 2005; Kanerva 2009). Die genannten Modelle sind dabei gleichzeitig die drei wohl bekanntesten Umsetzungen des Wortraums (siehe dazu unter anderem Turney/Pantel 2010). Im HAL-Modell wird nach Aufbau der Kookkurrenzmatrix die Varianz der Zeilen und Spalten errechnet, um anschließend nur die Merkmale mit der höchsten Varianz zu behalten. Während dieser Schritt

41 Sahlgren (2006) bezeichnet dieses Vorgehen als »directional«; bei einem bidirektionalen Vorgehen ergibt sich eine symmetrische Matrix, deren Zeilen und Spalten jeweils die gleichen Werte enthalten.

42 Neben dem Einfluss auf die Art der Ähnlichkeit spielt hierbei vor allem auch der Faktor der Dimensionalität bzw. der Vektorlänge eine große Rolle. Da in der Anwendung des Modells oftmals eine Vielzahl von Vektorvergleichen erforderlich ist (etwa bei einer Weiterverarbeitung mittels Clusteranalysen), ist ein möglichst kleines Merkmalsset beinahe unumgänglich. In vielen Arbeiten wird deshalb oftmals schon bei der Merkmalsauswahl von Methoden der Dimensionsreduktion gesprochen.

43 Dennoch schneiden in den Vergleichsstudien von Levy/Bullinaria (2001) die auf Basis der Frequenz verkürzten Vektoren sehr gut ab. Das ist vor allem insofern überraschend, weil unter den hochfrequenten Wörtern auch sehr viele Funktionswörter sind, die in der Regel als weitgehend neutral angesehen werden und deshalb in vielen Ansätzen ausgeklammert werden.

im HAL-Modell nur bei Bedarf eingesetzt wird, ist die Dimensionsreduktion im Modell des LSA ein inhärenter Bestandteil der Methodik. Hier wird die Dimensionsreduktion mittels Singular Value Decomposition (SVD) durchgeführt, einer speziellen Form der Hauptkomponentenanalyse. Neben einer Verkleinerung der Vektoren wird dadurch auch erreicht, dass neben den tatsächlichen Kookkurrenzen auch das Auftreten in ähnlichen Kontexten erfasst wird, so dass ›latente‹ Beziehungen aufgedeckt werden. Ebenso wie das HAL-Modell setzt auch die LSA zunächst den Aufbau einer vollständigen Matrix voraus. Im Gegensatz dazu geht das RI von vornherein von einem stark reduzierten Vektorraum aus. Zunächst wird für jedes Wort ein eindeutiger Indexvektor fester Länge erstellt (in der Regel wenige tausend Dimensionen), der an einigen wenigen, zufällig gewählten Positionen mit 1 und –1 belegt wird, ansonsten jedoch nur Nullen enthält.[44] Beim Durchlaufen des Korpus werden nun für jedes Wort die Indexvektoren sämtlicher Kookkurrenten innerhalb eines festgelegten Fensters hinzuaddiert. Analog zu den ›herkömmlichen‹ Kookkurrenzvektoren werden die Wörter auch hier durch ihr Verwendungsmuster repräsentiert, so dass die resultierende Kookkurrenzmatrix grundsätzlich die gleichen Eigenschaften aufweist wie bisher. Der wesentliche Unterschied liegt in der Beschaffenheit der Merkmale, die hier nicht für spezifische Kookkurrenten stehen, sondern eher eine Art verteilte Repräsentation darstellen.

Gewichtung

Neben der Wahl der Kontextbreite und der Merkmalsauswahl lassen sich die Wortvektoren auch beeinflussen, indem die Vektorelemente gewichtet werden. Eine sehr einfache Form der Gewichtung findet sich zum Beispiel im HAL-Modell. Die Gewichtung der Vektorelemente erfolgt hier umgekehrt proportional zum Abstand der Wörter zu einem gegebenen Zielwort: Beginnend mit dem Nachbarn werden absteigende Werte vergeben. Dieses Vorgehen ist in Abb. 4.5 anhand des Beispielsatzes »the horse raced past the barn« für eine Kontextbreite von fünf Elementen dargestellt.

In einem direktionalen[45] Vorgehen werden nur die Kookkurrenzen rechts des Zielworts in die Spalten eingetragen; die Zeilen der Kookkurrenzmatrix enthalten damit die Kookkurrenzen links des Zielworts. Bei der Erstellung der Wortvektoren werden im HAL-Modell Zeile und Spalte kombiniert, so dass die Vektorlänge der zweifachen Größe des Vokabulars entspricht. Die zusammengesetzten Vektoren enthalten damit die zusätzliche Information, auf welcher Seite die Kontextwörter auftreten.[46] Eine weitere Möglichkeit ist die textstatistische Bewertung

44 Mit der Verwendung von Zufallsvektoren nimmt das RI eine Sonderstellung gegenüber den anderen in diesem Abschnitt vorgestellten Modellen ein.

45 Siehe dazu die Ausführungen zum Kontextfenster in Anm. 41.

46 Anders als im Beispiel wird im HAL-Modell ein satzübergreifendes Kontextfenster der Breite 10 verwendet. Nach Burgess u.a. (1998, 6) wird dadurch der Einfluss rein syntaktischer Informationen minimiert: »As a further move away from dependence on syntax (or any structuring of the

	barn	horse	past	raced	the
barn		2	4	3	6
horse					5
past		4		5	3
raced		5			4
the		3	5	4	2

Abbildung 4.5: Nachbarschafts-Gewichtung im Hyperspace Analogue to Language (HAL). Die Gewichtung erfolgt umgekehrt proportional zum Zielwort und spiegelt damit die Nähe zum Zielwort wider (Beispiel nach Burgess u.a. 1998, 7).

der Vektorelemente, um eine zusätzliche Betonung bestimmter Kriterien vorzunehmen, etwa durch Übertragung von Maßen aus dem Information Retrieval wie zum Beispiel dem tf.idf-Maß.[47] Grundgedanke ist, dass einige Terme stärker diskriminieren, das heißt eine bessere Unterscheidung ermöglichen, da sie über einen höheren Informationsgehalt verfügen. Zum tf.idf-Maß gibt es eine Reihe von Alternativen, etwa Maße auf Basis der Termverteilung (sogenannte Term-Distribution-Models, siehe dazu Manning/Schütze 1999), oder auch Assoziationsmaße wie zum Beispiel die Log-Likelihood-Ratio nach Dunning (1993) oder die Mutual Information nach Church/Hanks (1990).[48]

Linguistische Informationen

Als Erweiterung der rein kookkurrenzbasierten Ansätze, die nur das vorhandene, unstrukturierte Vokabular als Merkmale einsetzen (sogenannte *bag-of-*

language under consideration other than that given by the division of words), sentence boundaries are ignored.«

47 Im tf.idf-Maß wird die Termfrequenz (tf) ins Verhältnis gesetzt zur sogenannten inversen Dokumentenfrequenz (idf), hier wiedergegeben nach Manning u.a. (2008):

$$w_{t,d} = (1 + \log\ \mathrm{tf}_{t,d}) \cdot \log \frac{N}{\mathrm{df}_t}$$

Die Termfrequenz (tf) entspricht der Häufigkeit des Auftretens eines Terms je Dokument, wobei die Häufigkeit in der Regel nicht einfach gezählt, sondern logarithmisch ›geglättet‹ wird – andernfalls würde das dreimalige Auftreten als dreifache Relevanz gegenüber einfachem Auftreten gewertet. Die Dokumentenfrequenz (df) bezeichnet dagegen die Anzahl der Dokumente, in denen der Term auftritt. Die idf setzt dies ins Verhältnis zur Gesamtanzahl der verfügbaren Dokumente, bezeichnet durch *N*. Grundgedanke ist hier, dass Terme, die nur in einem kleinen Teil der Dokumente auftreten, für diese Dokumente eine wichtigere Rolle spielen. Bei einer Übertragung des tf.idf-Maßes in den Wortraum wird df ersetzt durch die Anzahl der Verwendungskontexte, tf entspricht den jeweiligen Kookkurrenzwerten.

48 Für einen Überblick über verschiedene Maße und deren Eigenschaften siehe u.a. Evert (2005).

words-Modelle), können durch komplexere Vorverarbeitungsschritte wie zum Beispiel syntaktisches Parsing zusätzlich auch linguistische Informationen in das Modell aufgenommen werden, etwa grammatische Informationen wie Dependenz zur Unterscheidung von Subjekt- und Objektpositionen (siehe zum Beispiel Padó/Lapata 2007). Beispiele für die Verwendung von Strukturinformationen finden sich unter anderem bei Grefenstette (1994), Ruge (1992; 1995), Dagan u.a. (1993a; 1993b), Ansätze auf Grundlage der Filterung anhand syntaktischer Muster finden sich zum Beispiel bei Hearst (1992), Pennacchiotti/Pantel (2009), Almuhareb/Poesio (2004), oder Widdows/Dorow (2002). Ob solche syntaktisch motivierten Merkmale tatsächlich besser sind als reine Kookkurrenzen, ist nach Schütze/Pedersen (1997) zumindest zweifelhaft. Bei linguistisch motivierten Ansätzen zur Merkmalsauswahl muss man deshalb stets zwischen dem erzielten Mehrwert und der erhöhten Komplexität des Modells abwägen. So beinhalten die zusätzlichen Parameter, die man ins Modell einführt, das Risiko einer Verstetigung von eventuell fehlerhaften und damit verzerrenden Informationen.

4.1.3 Kontextvektoren und Kookkurrenzen zweiter Ordnung

Die im vergangenen Abschnitt beschriebenen Kookkurrenzvektoren spiegeln in einfacher Weise die Verwendungsmuster von Wörtern wider, so dass sich auf dieser Grundlage Ähnlichkeiten zwischen Wörtern bestimmen lassen. Jedoch geht dabei eine wichtige Information verloren: dadurch, dass die Summe aller Gebrauchskontexte zu einem einzigen Verwendungsmuster zusammengefasst wird, sind Kookkurrenzvektoren nicht in der Lage, zwischen verschiedenen Verwendungsweisen eines Wortes zu differenzieren.

Tatsächlich sind die Kookkurrenzvektoren in der ursprünglichen Konzeption von Schütze (1992; 1998) eigentlich nur ein erster Schritt. Das eigentliche Ziel ist die Repräsentation von einzelnen Gebrauchskontexten, um dadurch zu einer reichhaltigeren Repräsentation zu gelangen, die es ermöglicht, die Gemeinsamkeiten und Unterschiede zwischen einzelnen Verwendungsweisen zu erfassen. Um dies zu erreichen, nutzt Schütze die Kookkurrenzvektoren als Grundlage für die Repräsentation einzelner Gebrauchskontexte eines Wortes durch sogenannte *context vectors*.[49] Diese Kontextvektoren werden erstellt, indem der Durchschnitt (der sogenannte »Zentroid« bzw. Schwerpunkt) aus den Vektoren der Kontextelemente gebildet wird, schematisch dargestellt in Abb. 4.6.

Die Abbildung zeigt einen schematischen Vektorraum mit den zwei Dimensionen LEGAL und CLOTHES, in dem ein einzelnes Vorkommen des Wortes

49 Die Terminologie ist in diesem Bereich nicht immer einheitlich: so spricht auch Sahlgren (2006) von *context vectors*, jedoch bezieht er sich damit auf die oben beschriebenen einfachen Kookkurrenzvektoren.

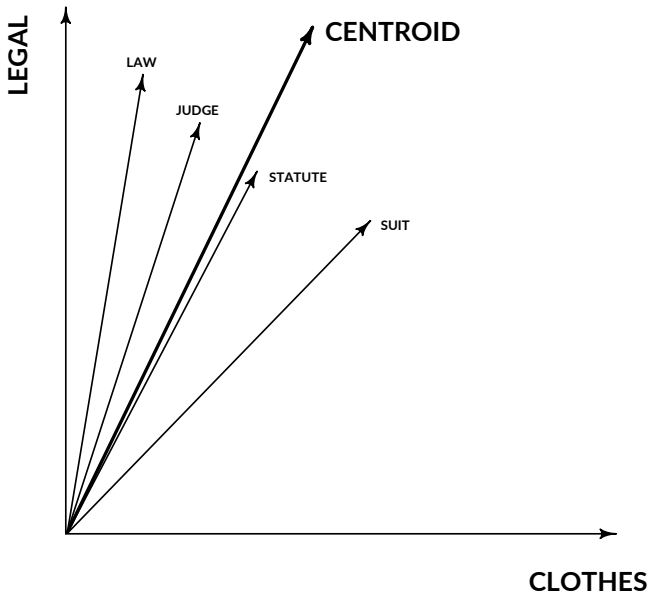


Abbildung 4.6: Der Kontextvektor für ein einzelnes Vorkommen von SUIT entspricht dem Schwerpunkt (CENTROID) der Vektoren der im Kontext auftretenden Wörter (Abbildung nach Schütze 1998).

SUIT dargestellt werden soll. Neben dem Wortvektor für SUIT werden auch die Vektoren der im Kontext auftretenden Wörter mit in den Raum projiziert. Der Kontextvektor wird erstellt, indem der Schwerpunkt (CENTROID) aller Vektoren errechnet wird. Während SUIT selbst Ähnlichkeiten zu beiden Dimensionen aufweist, wird der Schwerpunkt durch die beteiligten Kontextwörter in eine andere Richtung ›gezogen‹:

Neben den Kookkurrenten des betrachteten Worts selbst werden hier auch die Kookkurrenzen der Kookkurrenten für die Repräsentation genutzt. Dies bezeichnet Schütze (1998) als Kookkurrenz zweiter Ordnung, in Abgrenzung zu den oben beschriebenen Wortvektoren, die auf Kookkurrenz erster Ordnung basieren.[50] Wesentlicher Unterschied zu den einfachen Kookkurrenzvektoren ist der höhere Informationsgehalt der Repräsentation, in die auch die Verwendungsmuster der jeweiligen Kookkurrenten einfließen.

50 Weil in den Repräsentationen auch indirekte, nur über die Wortvektoren der Kookkurrenten verfügbare Information genutzt wird, spricht Schütze (1992, 2) hier in Anlehnung an konnektionistische Ansätze auch von »sublexikalischen« Repräsentationen.

Da die Kontextvektoren in den gleichen Wortraum projiziert werden wie zuvor die einfachen Kookkurrenzvektoren, sind Ähnlichkeitsvergleiche in gleicher Weise möglich, sowohl zwischen Kontext- und Wortvektoren als auch der Kontextvektoren untereinander. Im Vergleich zu den einfachen Kookkurrenzvektoren ist die Verwendung von Kookkurrenzen zweiter Ordnung weit weniger populär, was unter anderem mit dem hohen Verarbeitungs- und Speicheraufwand zusammenhängt, der sich aus der mehrfachen Repräsentation einzelner Wörter sowie der hohen Belegungsdichte der Repräsentationen ergibt. Der Hauptgrund ist jedoch, dass für Anwendungen, bei denen eine Differenzierung verschiedener Verwendungsweisen nicht entscheidend ist, bereits mit einfachen Kookkurrenzvektoren sehr gute Ergebnisse erzielt werden können, so dass die Vorteile der einfacheren Erstellung genutzt werden können.

4.1.4 Zusammenfassung

In den vorhergehenden Abschnitten wurde die technische Konzeption des WSM vorgestellt, das verschiedene Möglichkeiten zur Repräsentation von Wörtern bzw. sprachlichen Einheiten als Vektoren in hochdimensionalen Räumen bietet. Dadurch dass die Vektoren auf Grundlage der jeweiligen Gebrauchskontexte erstellt werden, spiegeln sie die Verwendungsmuster der repräsentierten Einheiten wider. Die Struktur des Wortraums ist dabei nicht fest vorgegeben; vielmehr beschreibt das Modell nur eine grundsätzliche Vorgehensweise, um Wörter auf Grundlage ihrer distributionellen Eigenschaften in Form von Vektoren zu repräsentieren.

Zentrale Parameter des Modells sind die Kontextbreite, die Gewichtung sowie die Merkmalsauswahl. Letzterer kommt eine besondere Rolle zu, da hiermit die Größe des verwendeten Vokabulars und damit auch die Dimensionierung des Vektorraums festgelegt wird. Im Kern geht es darum, die Merkmale so zu wählen, dass zum einen möglichst kurze Vektoren eingesetzt werden können, um damit den Verarbeitungsaufwand für Vektorvergleiche gering zu halten. Zum anderen müssen die Merkmale ein möglichst hohes diskriminatives Potential haben, das heißt eine möglichst zuverlässige Unterscheidung ermöglichen. Zusätzlich zur Merkmalsauswahl, die in der Regel schon vor der Erstellung erfolgt, kann die dem Vektorraum zugrunde gelegte Kookkurrenzmatrix auch nachträglich in ihrer Dimensionalität reduziert werden. Zwischen den Parametern und der Art des Vektorraums besteht ein direkter Zusammenhang, so dass sich anhand der Parametrisierungen verschiedene Instanzen des WSM unterscheiden lassen. Zu den bekanntesten Ausprägungen zählen unter anderem die Latent Semantic Analysis (Deerwester u.a. 1990), das Modell des Hyperspace Analogue to Language (Lund/Burgess 1996) oder auch das Random Indexing (Karlgren/Sahlgren 2001). Eine Übersicht über Eigenschaften und Potentiale des Wortraum-Modells geben unter anderem Turney/Pantel (2010) sowie Sahlgren (2006).

Die meisten Arbeiten zum Wortraum konzentrieren sich auf einfache Kookkurrenzvektoren. Von besonderem Interesse für diese Arbeit sind jedoch vor allem die »context vectors« nach Schütze (1992; 1998), mit denen einzelne Gebrauchskontexte repräsentiert werden können. Die Kontextvektoren basieren auf Kookkurrenzen zweiter Ordnung, bei denen auch die Kookkurrenzen der Kookkurrenten mit in die Repräsentation einfließen. Die resultierenden Vektoren sind dadurch deutlich reichhaltiger und können unter anderem dafür eingesetzt werden, zwischen verschiedenen Verwendungsweisen eines Wortes zu unterscheiden. Diese Möglichkeit zur Repräsentation einzelner Kontexte dient in dieser Arbeit als wesentliche Grundlage für die Modellierung der Bedeutungskonstitution (siehe Kapitel 5).

Nachdem das Modell bisher aus einer rein technischen Sicht erläutert wurde, werden im Folgenden die mit dem WSM verbundenen theoretischen Implikationen näher betrachtet. Dabei steht vor allem die Frage im Mittelpunkt, wie sich der Bedeutungsbegriff, der sich aus diesen Implikationen ergibt, zu den zuvor eingeführten Annahmen der Kognitiven Semantik verhält und inwieweit das WSM auch für die Modellierung eines dynamischen Bedeutungsbegriffs eingesetzt werden kann.

4.2 Theoretische Grundlagen des Modells

Wie im vergangenen Kapitel erläutert, werden im WSM die Verwendungsmuster von Wörtern in Form von Vektoren repräsentiert, die auf Grundlage einer statistischen Analyse ihrer sprachlichen Umgebungen erstellt werden. Das WSM ist jedoch nicht einfach ein Modell für die Ermittlung von Verwendungsähnlichkeiten; vielmehr erheben die verschiedenen Ausprägungen des Modells in der Regel den Anspruch, ein »computational model of meaning« (Sahlgren 2006, 17) zu sein, also ein verarbeitungstechnisch motiviertes Modell zur Repräsentation sprachlicher Bedeutung. Gleichzeitig ist das WSM auch ein Modell dafür, wie diese Repräsentationen erstellt werden. Neben einer eigenen »theory of representation« beinhaltet es demnach auch eine »theory of acquisition« (Sahlgren 2006, 17). Für beides (Repräsentation und Aufbau) gilt, dass erst eine Ausdeutung durch entsprechende Vorannahmen das WSM zu einem eigenständigen Modell der Bedeutungsrepräsentation und des-erwerbs macht.

In diesem Kapitel steht die Frage im Mittelpunkt, wie sich der mit dem Modell verbundene Bedeutungsbegriff zu den zuvor eingeführten Annahmen der Kognitiven Semantik verhält. Im Folgenden werden hierfür zunächst die beiden für den Bedeutungsbegriff maßgeblichen Vorannahmen des Modells, namentlich die »geometrische Metapher« (Abschnitt 4.2.1) und die »distributionelle Hypothese« (Abschnitt 4.2.2) erläutert und vor dem Hintergrund des in Kapitel 3 beschriebenen dynamischen Bedeutungsbegriffs eingeordnet. Auf dieser Grundlage wird in Abschnitt 4.2.3 diskutiert, wie das Modell in Verbindung mit kognitiv motivierten Annahmen eingesetzt werden kann.

4.2.1 Der Word Space als semantischer Raum

Die erste grundlegende Vorannahme betrifft die Art und Weise, wie Bedeutung im Modell repräsentiert wird und wie diese Repräsentationen interpretiert werden können: Wörter (bzw. allgemeiner: sprachliche Einheiten) werden im WSM auf Grundlage ihrer Verwendungsmuster in einem hochdimensionalen Vektorraum dargestellt; dieser wird als ein semantischer Raum verstanden, in dem die Vektoren als Wortbedeutungen und ihre Distanzen zueinander als Bedeutungsähnlichkeiten angesehen werden können. Sahlgren (2006) formuliert diese Sichtweise in Form einer Metapher:

> **The geometric metaphor of meaning:** *Meanings are locations in a semantic space, and semantic similarity is proximity between the locations.* (Sahlgren 2006, 19; Hervorhebung gemäß Original)

Mit der Interpretation durch eine geometrischen Metapher bezieht sich Sahlgren ganz explizit auf die Arbeiten von Lakoff/Johnson (1980; 1999), die in ihrem *conceptual metaphor approach* davon ausgehen, dass ein wesentlicher Teil unseres Denkens und Sprechens über die Welt, insbesondere auch über abstrakte Konzepte, von Metaphern bestimmt ist. Ausgehend von der These des »Embodiment«, der zufolge unsere kognitiven Fähigkeiten untrennbar mit der Körperlichkeit unserer physischen Existenz verbunden sind, sehen Lakoff und Johnson einige der grundlegendsten Metaphern in den räumlichen Aspekten dieser Körperlichkeit begründet (vergleiche Lakoff/Johnson, 1980). Vor diesem Hintergrund erscheint die hier eingesetzte geometrische Metapher, bei der räumliche Nähe genutzt wird, um das abstrakte Konzept semantischer Ähnlichkeit zu beschreiben, als intuitive und natürliche Wahl.[51]

Für die geometrische Metapher lässt sich eine deutliche Parallele zur Kognitiven Linguistik herstellen. Die Vorstellung einer räumlichen Interpretation von Bedeutungsähnlichkeit findet sich in verschiedener Ausprägung in vielen Arbeiten aus dem Bereich der Kognitiven Semantik bzw. allgemeiner der Kognitionswissenschaften wieder, etwa im Konzept der »Mental Spaces« nach Fauconnier (1994) oder den »Conceptual Spaces« nach Gärdenfors (2004; 2014), wobei als bekanntestes Beispiel neben den oben genannten Arbeiten von Lakoff hier sicherlich die Prototypentheorie nach Rosch (1975; 1978) zu nennen ist. Auch die in Abschnitt 3.2 vorgestellte Konzeption nach Cruse (2011) bzw. Croft/Cruse (2004) weist mit dem *conceptual space* eine entsprechende Analogie zur Raummetapher auf. Diese Analogie ist allerdings zunächst noch mit Vorsicht zu genießen: Es ist zwar die gleiche Metapher, mit der auch im konzeptuellen Raum räumliche Nähe als semantische Ähnlichkeit interpretiert wird; jedoch weicht die zugrunde gelegte Konzeption von der des semantischen Raums ab. Während Sahlgrens Formulierung der geometrischen Metapher davon ausgeht, dass sich Bedeutungen im

51 In direkter Anwendung der Metapher könnte man dies auch als ausgesprochen naheliegend bezeichnen.

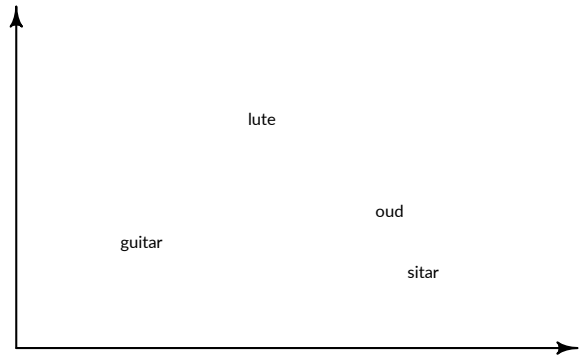


Abbildung 4.7: Geometrische Darstellung von semantischer Ähnlichkeit (Abbildung nach Sahlgren 2006, 18). In diesem schematischen Vektorraum können *oud* (eine Kurzhalslaute) und *sitar* aufgrund ihrer Nähe als ähnlicher zueinander als zu *guitar* interpretiert werden.

semantischen Raum lokalisieren lassen, können es im konzeptuellen Raum aus Sicht der Kognitiven Semantik nicht die Bedeutungen selbst sein, die zueinander in Beziehung gesetzt werden, da dies der Konzeption eines dynamischen Bedeutungsbegriffs zuwiderliefe. Inwieweit sich der semantische Raum des WSM dennoch auch für einen kognitiv orientierten Bedeutungsbegriff erschließen lässt, hängt in erster Linie davon ab, was genau unter den von Sahlgren in der geometrischen Metapher genannten *meanings* zu verstehen ist, die im Wortraum repräsentiert sind. Hierfür wird im Folgenden zunächst erörtert, auf welcher Grundlage die Repräsentationen als Bedeutungen interpretiert werden, um daraus eine Möglichkeit zu entwickeln, wie sie in einer Weise ausgelegt werden können, die mit den Annahmen der Kognitiven Semantik vereinbar ist.

4.2.2 Die distributionelle Hypothese

Die zweite grundlegende Annahme bezieht sich auf die Art und Weise, wie die Repräsentationen erstellt werden: Das Modell stützt sich auf die These, dass die Verwendungsmuster, die aus Kookkurrenzen in Textkorpora gewonnen werden, als Basis für die Repräsentation sprachlicher Bedeutung dienen können. Grundlage für diese Annahme ist die sogenannte distributionelle Hypothese (DH), hier zitiert nach Sahlgren (2006):

> **The distributional hypothesis:** *words with similar distributional properties have similar meanings*. (Sahlgren 2006, 21; Hervorhebung gemäß Original)

Sahlgren sieht den Ursprung der DH vor allem im amerikanischen Strukturalismus und bezeichnet sie dementsprechend als »rooted in structuralist soil«

(Sahlgren 2008, 34).[52] Die wesentliche Grundlage sieht Sahlgren in der distributionellen Methodik nach Zellig Harris, dessen erklärtes Ziel die Entwicklung einer umfassenden Methode zur linguistischen Analyse war, um der Linguistik als Wissenschaft eine klar umrissene Grundlage zu geben. Zentraler Gedanke der distributionellen Methodik ist es, dass die grundlegenden Einheiten von Sprache (Phoneme und Morpheme, aber auch syntaktische Einheiten) allein auf Basis ihrer distributionellen Eigenschaften in Klassen organisiert und dadurch zueinander in Beziehung gesetzt werden können. Zwar hebt Sahlgren hervor, dass sich bei Harris selbst eigentlich keine explizite semantische Konzeption findet, jedoch findet er Hinweise, dass die Methodik auch auf Bedeutung angewendet werden kann. Das wesentliche Argument ist, dass Harris seine distributionelle Methode als vollständig in Bezug auf linguistische Phänomene ansieht und dass es somit auch möglich sein muss, sprachliche Bedeutung zum Gegenstand der Analyse zu machen. Bestätigt sieht er dies vor allem durch das folgende Zitat:

> If we consider words or morphemes A and B to be more different in meaning than A and C, then we will often find that the distributions of A and B are more different than the distributions of A and C. In other words, difference in meaning correlates with difference of distribution. (Harris 1954, 156)

Harris sieht demnach eine Korrelation zwischen semantischen und distributionellen Unterschieden. Dies impliziert nach Sahlgren, dass die distributionelle Methodik auch als »discovery procedure« für die Aufdeckung semantischer Beziehungen genutzt werden kann (Sahlgren 2008, 36). Ganz wesentlich ist dabei die Abgrenzung gegenüber außersprachlichen Faktoren: Die Distribution dient als alleinige Quelle der Information – nur die Aspekte sprachlicher Bedeutung, die tatsächlich in Sprache enthalten sind, fließen in die Analyse mit ein.

Empirische Unterstützung findet die DH nach Sahlgren unter anderem durch Rubenstein/Goodenough (1965), sowie darauf aufbauend bei Miller/Charles (1991), die in einem Vergleich der kontextuellen Ähnlichkeiten von Synonymen mit den Bewertungen von Probanden Evidenz für die Korrelation zwischen semantischer und kontextueller Ähnlichkeit finden. Über die reine Bestätigung der DH hinaus sehen Miller und Charles in den kontextuellen Repräsentationen abstrakte kognitive Strukturen auf Grundlage der tatsächlichen (und potentiellen) Verwendungen des Wortes. Miller und Charles stützen sich dabei

52 Die Sicht, dass ein enger Zusammenhang zwischen Gebrauch und Bedeutung besteht, steht in einer längeren Tradition gebrauchsorientierter Ansätze. Häufig wird in diesem Zusammenhang auch auf die Arbeiten von J.R. Firth zum Kontextualismus verwiesen, insbesondere auf das Zitat »You should know a word by the company it keeps« (Firth 1957, 11). In diesem Zusammenhang wird ebenfalls sehr häufig Ludwig Wittgenstein zitiert, der in §43 der »Philosophischen Untersuchungen« schreibt: »Die Bedeutung eines Wortes ist sein Gebrauch in der Sprache« (Wittgenstein 1953) – zumeist jedoch ohne tatsächlich näher auf das Zitat einzugehen (so auch hier).

ganz wesentlich auf die Beobachtung, dass wir die Bedeutung vieler Wörter ausschließlich auf Grundlage des Kontextes erlernen, ohne tatsächliche persönliche Erfahrung mit den sie bezeichnenden Dingen oder Gegebenheiten zu haben.

Während sich für die geometrische Metapher durchaus eine Parallele zur Kognitiven Semantik herstellen lässt (siehe Abschnitt 4.2.1), so ist dies in Bezug auf die DH nicht ohne weiteres möglich. Zwar markiert die mit ihr verbundene distributionelle Methodik das Modell als gebrauchsorientierten Ansatz,[53] was für sich genommen einen klaren Berührungspunkt zur Kognitiven Semantik darstellt; jedoch ist die Reduktion auf Distribution als alleinige Grundlage semantischer Repräsentationen zumindest problematisch. Das Problem entsteht jedoch erst, wenn die distributionellen Eigenschaften als konstitutiv für Bedeutung angesehen werden. Zwar ist es für die Kognitive Semantik durchaus akzeptabel, linguistisches Wissen aus rein linguistischer Erfahrung zu gewinnen. Der Unterschied besteht jedoch darin, dass die Kognitive Linguistik nicht akzeptiert, dass die Distribution die Grundlage für Bedeutung sein soll, sondern dass bei der Gewinnung von semantischem Wissen stets ein kognitiver Prozess involviert ist, der in der Konzeptualisierung von Bedeutung besteht. Gemeint ist damit, dass die Konzeptualisierung nicht einfach in der Abstraktion über Distributionen besteht, sondern vielmehr in der Verankerung dieser abstrakten Repräsentationen im konzeptuellen Raum, der nicht nur semantische Strukturen enthält, sondern das gesamte – also auch nichtsprachliche – Erfahrungswissen (siehe Abschnitt 3.2). Aus Sicht der Kognitiven Linguistik setzen Bedeutungen demnach stets eine kognitive Aktivität, also die Beteiligung von Sprechern voraus – im Wortraum existieren die Bedeutungen dagegen vermeintlich unabhängig in Form von abstrakten Mustern, die aus der Summe der Kontextualisierungen gewonnen werden.

4.2.3 Diskussion

Aus den vorangehenden Ausführungen ist deutlich geworden, auf welcher Grundlage das WSM den Anspruch erhebt, ein eigenständiges Modell für die Ermittlung und Repräsentation sprachlicher Bedeutung zu sein. Die durch die getroffenen Vorannahmen mit dem Modell verbundene Konzeption von Bedeutung weist dabei ganz offenkundig deutliche Unterschiede gegenüber dem in Abschnitt 3.2 beschriebenen kognitiv motivierten Bedeutungsbegriff auf. Die Unterschiede sind jedoch nicht unüberbrückbar: letztlich bestimmt erst die konkrete Auslegung der Distributionellen Hypothese (DH), welchen Stellenwert die distributionelle Methodik für den Bedeutungsbegriff bekommt – ob das WSM für die Modellierung eingesetzt werden kann, ist demnach vor allem eine Frage des Status, den man den

53 Das WSM vertritt hierin einen durch und durch deskriptiven Ansatz: Die Repräsentationen werden ausschließlich auf Grundlage von tatsächlichen Verwendungen in Korpora erstellt, ohne externe Informationen oder Eingriffe durch menschliche Bearbeiter.

distributionellen Eigenschaften für die Erklärung von Bedeutung auf kognitiver Ebene einräumt. Nach Lenci (2008) ist dies eine Frage der Auslegung der DH, die er in der folgenden Formulierung wiedergibt:

> The degree of semantic similarity between two linguistic expressions A and B is a function of the similarity of the linguistic contexts in which A and B can appear. (Lenci 2008, 3)

Diese Formulierung lässt gegenüber der von Sahlgren etwas mehr Spielraum für die Interpretation: nach Lenci besagt die DH zunächst nur, dass zumindest einige Aspekte der Wortbedeutung aus ihren distributionellen Eigenschaften erschlossen werden können. Mit dieser Formulierung wird zwar ebenfalls ein (funktionaler) Zusammenhang zwischen Distribution und Bedeutung hergestellt. Es ist jedoch nicht automatisch etwas darüber gesagt, ob man die funktionale Beziehung von Distribution und Semantik als korrelativ versteht oder als kausale Abhängigkeit. Diese beiden Auslegungen korrespondieren nach Lenci zu den »two souls« der DH (Lenci 2008, 13), für die er eine Unterscheidung zwischen starker und schwacher DH vorschlägt.

In ihrer Auslegung als ›starke DH‹ wird Distribution als konstitutiv für Bedeutung angesehen. Zwischen semantischem Gehalt und Distribution liegt damit eine kausale Beziehung vor: das distributionelle Verhalten dient hier der Erklärung des semantischen Gehalts auf kognitiver Ebene. Nach Lenci ist die DH in ihrer starken Auslegung somit eine ›kognitive‹ Hypothese über die Beschaffenheit semantischer Repräsentationen, die sich vor allem an der Sicht von Miller/Charles (1991) orientiert. Diese sind – neben Rubenstein/Goodenough (1965) – deshalb auch als typische Vertreter dieser Auslegung anzusehen. Nach Lenci basieren die meisten Implementationen des WSM auf der starken DH, so zum Beispiel auch die Latent Semantic Analysis nach Deerwester u.a. (1990) und das Modell des Hyperspace Analogue to Language nach Lund/Burgess (1996), was sich darin äußert, dass sie unmittelbar, das heißt ohne zusätzliche theoretische Annahmen, für die Modellierung psycholinguistischer Phänomene eingesetzt werden. Das Bekenntnis zur DH hat dem WSM in neueren Arbeiten den Beinamen Distributional Semantic Models (DSM) eingebracht, wobei die in aktuelleren Ansätzen oftmals propagierte Idee einer distributionellen Semantik erst in den letzten Jahren aufgekommen ist. Neben der Bezugnahme auf Harris wird hierbei häufig auch direkt auf Miller/Charles (1991) verwiesen, die der DH zusätzlich eine kognitive Relevanz zusprechen.

In der schwachen Auslegung nimmt die DH dagegen nur eine Korrelation zwischen semantischem Gehalt und Distribution an. Die distributionellen Eigenschaften spiegeln demnach zwar die semantischen Eigenschaften wider; sie werden jedoch nicht als konstitutiv für die Bedeutung angesehen. Distributionelle Eigenschaften werden in der schwachen Auslegung somit nicht als Ursache, sondern vielmehr als Ausdruck von semantischen Eigenschaften angesehen: Grundidee ist hier, dass die Wortbedeutung das kombinatorische Verhalten

bestimmt, unabhängig davon, wie sie genau definiert ist. Bedeutung ist hier eher eine Art »latenter Variable« (Lenci 2008, 14), welche die sichtbare Distribution bestimmt – und die über die quantitative Untersuchung der distributionellen Eigenschaften aufgedeckt werden soll. In der Auslegung als schwache DH wird das WSM demnach nur als methodischer Zugang gesehen, als quantitative Methode zur Untersuchung semantischer Eigenschaften vergleichbar zu Harris' distributioneller Analyse bzw. zu korpusbasierten Untersuchungen im Allgemeinen. Harris selbst schließt nicht aus, dass Bedeutung auch von extralinguistischen Faktoren abhängt, womit er sich unmittelbar an Bloomfield orientiert. Er betont jedoch, dass selbst in solchen Fällen ein sichtbares Pendant in der Distribution zu erwarten ist:

> As Leonard Bloomfield pointed out, it frequently happens that when we do not rest with the explanation that something is due to meaning, we discover that it has a formal regularity or ›explanation‹. It may still be ›due to meaning‹ in one sense, but it accords with a distributional regularity. (Harris 1954, 156)

Die distributionelle Methodik dient in dieser Perspektive dazu, den theoretischen Annahmen über Wortbedeutung ein robusteres empirisches Fundament zu geben, indem mittels einer Analyse der Distribution die zugrundeliegenden semantischen Eigenschaften aufgedeckt werden.

Während die konkreten Umsetzungen des WSM in der Regel einer starken Auslegung der DH verpflichtet sind, der zufolge distributionelle Eigenschaften als konstitutiv für Bedeutung angesehen werden, kann das Modell bei einer schwachen Auslegung der DH durchaus auch eingesetzt werden, ohne damit automatisch einen rein distributionellen Bedeutungsbegriff zu unterschreiben. In der schwachen Auslegung beschreibt die DH nur einen methodischen Ansatz, der selbst keinen spezifischen Bedeutungsbegriff beinhaltet und damit auch mit anderen Auslegungen des Bedeutungsbegriffs vereinbar ist – also auch mit dem in Abschnitt 3.2 skizzierten dynamischen Bedeutungsbegriff. Es läuft damit im Rahmen dieser Arbeit auf eine Art Arbeitsteilung heraus: das WSM liefert die Methodik, um distributionelle Unterschiede sichtbar zu machen; die Kognitive Semantik liefert die Interpretation dieser Unterschiede, indem sie diese als einen Reflex der zugrundeliegenden kognitiven Prozesse ansieht.

4.3 Zusammenfassung

In diesem Kapitel wurde das Word Space Model (WSM) als Grundlage für die Operationalisierung der in Kapitel 3 getroffenen Annahmen zur Bedeutungskonstitution vorgeschlagen. Das wesentliche Motiv für die Verwendung des WSM ist der Umstand, dass die Kognitive Semantik selbst keine einheitliche Methodik zur Ermittlung und Darstellung von Bedeutung bereitstellt. Das WSM bietet sich hier vor allem deshalb an, da es den von der Kognitiven Semantik propagierten

Fokus auf den Sprachgebrauch konsequent umsetzt. So markiert die distributionelle Methodologie das Modell als gebrauchsorientierten Ansatz, mit der wichtigen Einschränkung auf schriftliche bzw. verschriftlichte Sprache in Textkorpora, wobei keine andere Information als die in den Texten enthaltene verwendet wird. Diese ›Vereinfachung‹[54] bringt einen wesentlichen Vorteil mit sich: Der korpuslinguistische Zugang ermöglicht es, die in Sprache enthaltenen Strukturierungseigenschaften zu nutzen, ohne dass diese explizit – etwa durch formale Beschreibung – vorliegen.

Der Verzicht auf zusätzliche Beschreibungssprachen und menschliche Eingriffe unterscheidet das Modell ganz wesentlich von den meisten der verbreiteten Repräsentationsformalismen wie semantischen Netzen (etwa WordNet[55] oder dessen deutscher Variante GermaNet[56]) oder auch der von Croft/Cruse (2004) für die Operationalisierung vorgeschlagenen Frame-Semantik nach Fillmore (1976; 1982) bzw. deren Umsetzung beispielsweise durch FrameNet[57]. Das WSM ist zudem nicht zuletzt aus rein verarbeitungstechnischer Sicht sehr attraktiv. Neben der Einfachheit und Kompaktheit des Modells ist dies vor allem auch in der resultierenden numerischen Repräsentation begründet, wie sie durch die Vektoren des Wortraums gegeben ist. Zum einen bedeutet diese einen geringen Grad an Formalisierung, zum anderen eröffnet eine numerische Repräsentation die Möglichkeit zur Nutzung von mathematischen Vergleichsmetriken, was neben einer einfachen und intuitiven Operationalisierung von semantischer Ähnlichkeit durch räumliche Nähe auch den Einsatz von gängigen Clusteranalysen und Klassifikationsverfahren ermöglicht.

Die mit dem WSM verbundene Methodik bietet sich somit gleich aus mehreren Gründen für die Operationalisierung der Bedeutungskonstitution im Sinne der Kognitiven Semantik an. Um die im WSM realisierte distributionelle Methodik im Zusammenhang mit dem dynamischen Bedeutungsbegriff der Kognitiven Semantik verwenden zu können, muss jedoch von dem üblicherweise mit dem WSM verbundenen distributionellen Bedeutungsbegriff Abstand genommen werden. Grundlage für die Übertragung ist dabei eine schwache Auslegung der Distributionellen Hypothese (DH), die als zentrale Vorannahme des WSM das Verhältnis von Kontext und Bedeutung zum Gegenstand hat. Während die starke DH davon ausgeht, dass Distribution konstitutiv für Bedeutung ist und beides somit gleichzusetzen ist, besagt die DH in ihrer schwachen Auslegung nur, dass zwar ein enger Zusammenhang zwischen Distribution und Bedeutung besteht, dass dieser jedoch im Sinne einer Korrelation zu verstehen ist. Aus Sicht der Kognitiven Semantik ist es die Bedeutung (die Konzeptualisierung), welche die Distribution bestimmt, nicht umgekehrt. Auf dieser Grundlage können die Vektoren,

54 Sahlgren spricht hier von »simplifying assumptions«, siehe Sahlgren (2006, 12).
55 Siehe https://wordnet.princeton.edu (Zugriff vom 21.02.2018).
56 Siehe http://www.sfs.uni-tuebingen.de/GermaNet (Zugriff vom 21.02.2018).
57 Siehe https://framenet.icsi.berkeley.edu (Zugriff vom 21.02.2018).

die aus technischer Sicht im Grunde nur die Verwendungsmuster von Wörtern widerspiegeln, als ein Reflex der zugrundeliegenden semantischen Eigenschaften interpretiert werden. So können aus den distributionellen Eigenschaften zwar semantische Eigenschaften abgelesen werden, jedoch sind sie damit nicht restlos erklärt. Im Rahmen dieser Arbeit wird das WSM somit im Sinne einer methodischen Ergänzung für die Kognitive Semantik eingesetzt, ohne gleichzeitig einen rein distributionellen Bedeutungsbegriff zu übernehmen.

Vor dem Hintergrund einer schwachen Auslegung der DH lässt sich auch die geometrische Metapher neu bewerten. Wie bereits in Abschnitt 4.2.1 angeführt, kann die räumliche Nähe im Wortraum auch aus Perspektive der Kognitiven Semantik als semantische Ähnlichkeit gedeutet werden, und auch die Vorstellung, dass es sich bei den Vektoren des Wortraums um semantische Strukturen handelt, muss nicht vollständig zurückgewiesen werden. Dies gilt jedoch nur, wenn die semantischen Strukturen nicht mit Bedeutungen gleichgesetzt werden, da eine statische Repräsentation von Bedeutungen sich nicht mit dem dynamischen Bedeutungsbegriff der Kognitiven Semantik verträgt. Eine entsprechende Umdeutung wird im folgenden Kapitel vorgenommen, wenn die eigentliche Operationalisierung der Bedeutungskonstitution auf Grundlage des WSM beschrieben wird.

5. Bedeutungskonstitution im Vektorraum

Mit den Vektoren des Word Space Model (WSM) stehen nun die wesentlichen Bausteine für die Modellierung bereit. Zudem konnte durch eine schwache Auslegung der Distributionellen Hypothese die grundsätzliche Vereinbarkeit des WSM mit den Annahmen der Kognitiven Semantik aufgezeigt werden, insofern das WSM hier vor allem als ein methodischer Ansatz verstanden wird, der nicht zwingend einem rein distributionellen Bedeutungsbegriff verpflichtet ist. Auf dieser Grundlage soll in diesem Kapitel nun beschrieben werden, wie sich der von Cruse (2011) in seinem *dynamic construal approach* skizzierte Prozess der Bedeutungskonstitution über das WSM operationalisieren lässt, indem die dem Prozess zugrundeliegenden Konzepte auf das WSM abgebildet werden.[58] Wie in Abschnitt 3.3 dargelegt, ist die Kognitive Linguistik dabei nur als ein theoretischer Bezugsrahmen für die computerlinguistische Modellierung zu verstehen, nicht als abgeschlossene Theorie, die eine (womöglich gar wortgetreue) Umsetzung bzw. Operationalisierung vorzeichnen würde. Ganz im Gegenteil: vielmehr ist das Fehlen einer Operationalisierung ein zentrales Motiv dieser Arbeit. Die hier vorgenommene Modellierung versteht sich damit ganz explizit als Vorschlag, die distributionelle Methodik für die Kognitive Semantik zu erschließen. Cruses dynamic construal approach dient hierbei als konzeptionelle Vorlage, quasi als Leitbild für die Modellierung einer grundsätzlichen Konzeption einer Bedeutungskonstitution aus Sicht der Kognitiven Semantik.

In Abschnitt 5.1 wird auf Grundlage des WSM das Format für die Eingabe- und Ausgabeinformation des Prozesses spezifiziert. Anschließend wird in Abschnitt 5.2 der eigentliche Prozess der Bedeutungskonstitution als eine Transformation von Vektoren beschrieben, deren Ergebnis als lokale Bedeutung interpretiert werden kann. Die ermittelten lokalen Bedeutungen können zudem zueinander in Beziehung gesetzt werden, um dadurch im Sinne von Cruses Konzeption das volle semantische Potential eines Wortes zu erfassen. In Abschnitt 5.3 schließlich wird die hier vorgeschlagene Operationalisierung nochmals im Verhältnis zu den theoretischen Vorannahmen diskutiert.

5.1 Repräsentation von Input und Output

Gemäß Cruses *dynamic construal approach* lässt sich der Prozess der Bedeutungskonstitution in der Differenz von abstraktem Bedeutungspotential und konkreter Bedeutung verorten. Um dies auf Grundlage des WSM in ein

58 Gegenstand des Kapitels ist damit die Umsetzung einer abstrakten Verarbeitungstheorie (hier zur Erklärung der Variabilität von Bedeutung), was im Wesentlichen der zweiten Beschreibungsebene im Sinne von Marr (1982) entspricht (vgl. Abschnitt 1.2).

computerlinguistisches Modell überführen zu können, bedarf es hier zunächst einer angemessenen Repräsentation des Bedeutungspotentials durch Vektoren, auf deren Grundlage dann die Bedeutungskonstitution operationalisiert werden kann. Wie das Bedeutungspotential unmittelbar im WSM repräsentiert werden kann, wird in Abschnitt 5.1.1 beschrieben. Auf dieser Grundlage wird anschließend die Ein- und Ausgabeinformation für den Prozess spezifiziert (Abschnitt 5.1.2), die aufseiten des Inputs neben einer vektoriellen Repräsentation des Bedeutungspotentials auch den Kontext umfasst und infolge des Prozesses wiederum in einem einzelnen Vektor resultiert.

5.1.1 Bedeutungspotential im Vektorraum

Für die Repräsentation des Bedeutungspotentials im WSM muss zunächst eine Umdeutung der Kookkurrenzvektoren vorgenommen werden. Werden diese im WSM zumeist als eine unmittelbare Repräsentation von Wortbedeutungen interpretiert (etwa in der Auslegung des WSM als sogenannte Distributional Semantic Models), so ist dies aus Perspektive der Kognitiven Semantik streng genommen nicht möglich, da es sich nicht mit einem dynamischen Bedeutungsbegriff verträgt, bei dem sich die Bedeutung erst in der konkreten Verwendung auf der konzeptuellen Ebene konstituiert. Mittels einer schwachen Auslegung der Distributionellen Hypothese, wie sie in Abschnitt 4.2.3 diskutiert wurde, können die Vektoren des WSM jedoch durchaus auch für die Modellierung einer kognitiv motivierten Bedeutungskonzeption eingesetzt werden – allerdings nur unter der Prämisse, dass die Kookkurrenzvektoren eben gerade nicht als vollwertige Bedeutungen angesehen werden, sondern vielmehr nur als vorläufige Strukturen. Deshalb werden die Kookkurrenzvektoren in dieser Arbeit stattdessen als Repräsentationen der Bedeutungspotentiale ausgelegt. Das Bedeutungspotential umfasst in der Konzeption von Cruse zwei Bestandteile: zum einen den *purport*, mit dem Cruse einen »body of conceptual content« bezeichnet (siehe Croft/Cruse 2004, 100), das heißt eine (unbestimmte) Menge an konzeptuellem Gehalt, der die Bedeutungsmöglichkeiten in Abgrenzung zu anderen Wörtern bestimmt; zum anderen eine Reihe von *conventionalized constraints*, welche im Sinne von sprachlichen Konventionen die Verwendungsmöglichkeiten eingrenzen.

Tatsächlich lässt sich beides unmittelbar in den Kookkurrenzvektoren des Wortraums verorten: da diese auf Grundlage konkreter Verwendungen erstellt werden, enthalten sie stets das vollständige Verwendungsmuster. Da sie damit die oftmals heterogenen Kontexte in einer einzigen Repräsentation zusammenfassen, sind sie in Bezug auf die (potentiell verschiedenen) Bedeutungen der repräsentierten Wörter zunächst nicht weiter ausgedeutet. Das Verwendungsmuster soll im Kontext dieser Arbeit somit als »purport« ausgelegt werden, das heißt als eine (unbestimmte) Menge an konzeptuellem Gehalt, der die Bedeutungsmöglichkeiten des Wortes eingrenzt. Ebenfalls in den Kookkurrenzvektoren kodiert

sind die allgemeinen Verwendungseigenschaften der jeweils repräsentierten Wörter. Sie spiegeln die kombinatorischen Möglichkeiten direkt wider und enthalten damit implizite Strukturinformationen darüber, in welchen Kontexten ein Wort typischerweise auftritt. Diese Informationen lassen sich in der Übertragung als *conventionalized constraints* interpretieren, insofern die Repräsentation festlegt, unter welchen Bedingungen ein Wort verwendet werden kann. Daraus ergibt sich das in Abb. 5.1 wiedergegebene Schema.

Mit dieser Umdeutung können die Kookkurrenzvektoren als Ausdruck der allgemeinen semantischen Eigenschaften der Wörter, das heißt ihres semantischen Potentials angesehen werden. In dieser Arbeit dienen die Kookkurrenzvektoren somit nicht als Grundlage für die Repräsentation von Bedeutungen, sondern vielmehr der Bedeutungspotentiale – denn Bedeutung trägt in kognitiver Perspektive nur das kontextualisierte Wort.

5.1.2 Input und Output als Vektoren

Auf Grundlage der Umdeutung der Kookkurrenzvektoren des Wortraums lässt sich nun spezifizieren, wie die Ein- und Ausgabeinformation in der Modellierung repräsentiert werden kann. Die vollständige Eingabeinformation für den Prozess der Bedeutungskonstitution besteht neben dem Bedeutungspotential eines Zielworts, das durch seinen Kookkurrenzvektor repräsentiert wird, zusätzlich

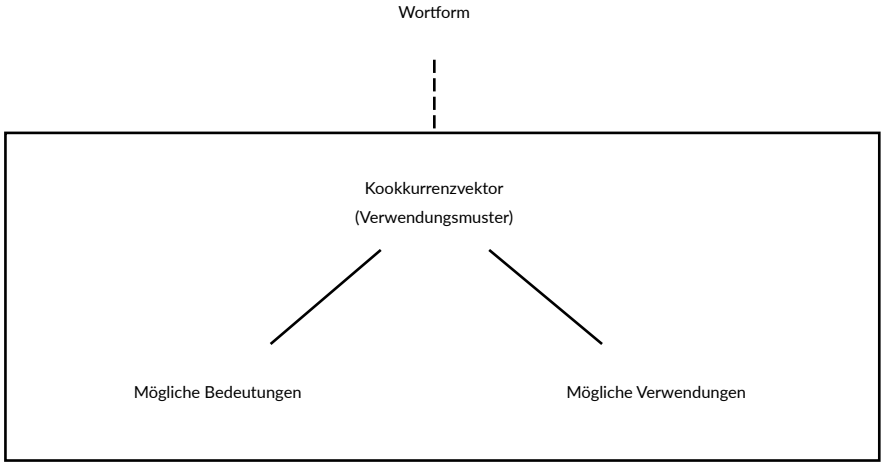


Abbildung 5.1: Differenzierung des Bedeutungspotentials, übertragen in das WSM. Das Bedeutungspotential wird durch einen Kookkurrenzvektor repräsentiert, der das Verwendungsmuster eines Wortes widerspiegelt. Das Muster enthält die Verwendungsmöglichkeiten sowie implizit die möglichen Bedeutungen und repräsentiert damit sowohl den *purport* als auch die *conventionalized constraints*.

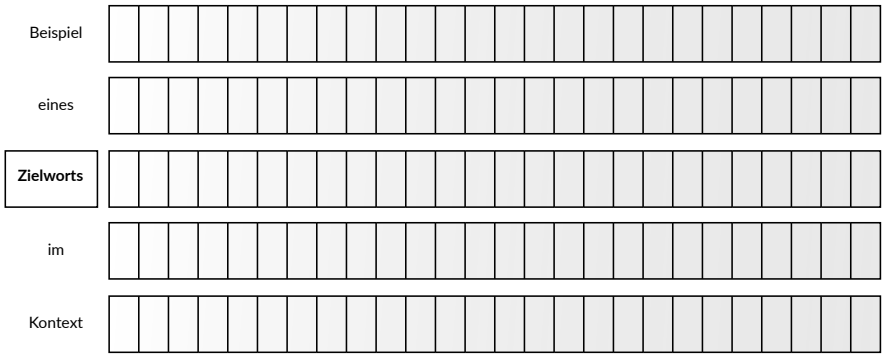


Abbildung 5.2: Das Eingabeformat für den Prozess. Sowohl das Zielwort (in der Abbildung markiert) als auch die in dessen Kontext auftretenden Wörter werden durch einfache Kookkurrenzvektoren repräsentiert. Die Anzahl der eingegebenen Vektoren wird durch die Kontextbreite festgelegt.

auch aus dem Kontext, in dem das Wort auftritt. In dieser Arbeit wird dabei eine Einschränkung vorgenommen, die sich unmittelbar aus der Wahl des WSM als Grundlage für die Repräsentation ergibt. Die mit dem WSM verbundene distributionelle Methodik impliziert eine Modellierung auf Grundlage von Korpora, was eine Beschränkung auf verschriftlichte Sprache und damit auch eine Verengung des Kontextbegriffs zur Folge hat: da bei der Beschränkung auf Korpusdaten keine andere als die in den Texten enthaltene Information verwendet wird, kann im Modell nur das linguistische Umfeld berücksichtigt werden – andere Formen der Kontextualisierung (etwa der physische oder soziale Kontext) sind somit ohne eine vorherige Repräsentation in der Modellierung nicht zugänglich. Der Kontext umfasst im Modell somit einfach die im Kontext des Zielworts auftretenden Wörter, die analog zum Zielwort ebenfalls durch ihre Kookkurrenzvektoren repräsentiert sind. Abb. 5.2 illustriert die vollständige Eingabeinformation des Prozesses.

Der Input besteht demnach aus einer Menge von Kookkurrenzvektoren, dem des Zielworts sowie denen der Kontextelemente.[59] Ergebnis des Prozesses ist eine lokale Bedeutung, die aus einer Transformation des eingegebenen Bedeutungspotentials resultiert. Ebenso wie der Input wird auch das Ergebnis des Prozesses durch einen Vektor repräsentiert. Im Zuge der Bedeutungskonstitution verändert sich zwar die Belegung des ursprünglichen Vektors, jedoch nicht seine Struktur. Da der Vektorraum, in dem die resultierenden Vektoren verortet sind, noch immer der gleiche ist wie zu Beginn des Prozesses, können die resultierenden Vektoren direkt mit ihrer Ausgangsstruktur verglichen werden, so dass die Abweichung

59 Die genaue Anzahl der eingegebenen Vektoren orientiert sich daran, wie breit der Kontext im Modell letztlich angesetzt wird. Für den Moment ist die Frage der Kontextbreite jedoch nicht wesentlich, da es hier zunächst nur um das Repräsentationsformat geht.

zwischen Input und Output auch räumlich abgebildet werden kann. Denn an der Sicht auf den Wortraum als semantischen Raum ändert die vorherige Umdeutung nichts: der Wortraum wird hier nach wie vor als semantischer Raum interpretiert, jedoch sind in diesem durch die einfachen Kookkurrenzvektoren nicht direkt die Bedeutungen repräsentiert, sondern zunächst nur die Bedeutungspotentiale, welche die möglichen Bedeutungen umfassen, ohne diese explizit zu machen. Räumliche Nähe kann dennoch noch immer als semantische Ähnlichkeit interpretiert werden; jedoch ist es hier nur die Ähnlichkeit zwischen Potentialen. Was für die Erfassung der konkreten Bedeutung nun noch fehlt, ist der Prozess der Bedeutungskonstitution auf Grundlage der Vektoren; dieser wird im Folgenden beschrieben.

5.2 Bedeutungskonstitution als Transformation von Vektoren

Mit der Umdeutung der Kookkurrenzvektoren steht nun ein Modell für die Repräsentation des Bedeutungspotentials zur Verfügung, auf dem der Prozess der Bedeutungskonstitution aufsetzen kann. Wie aber erfolgt nun der eigentliche Prozess? Und welche Bedingungen stellt die Wahl der Repräsentation an die algorithmische Beschreibung des Prozesses? Diese Fragen bilden den zentralen Gegenstand dieses Abschnitts.

Cruse beschreibt den Prozess der Bedeutungskonstitution als ein *construal*, im Zuge dessen das Bedeutungspotential in eine konkrete Bedeutung überführt wird (siehe dazu Abschnitt 3.2.2). Bei einer Operationalisierung auf Grundlage des WSM muss in Bezug auf den Begriff des *construal* hier eine Vereinfachung vorgenommen werden: aus Perspektive der Modellierung ist dieser viel zu voraussetzungsreich und muss deshalb in diesem Zusammenhang etwas schwächer ausgelegt werden. Wo Cruse den Prozess der Bedeutungskonstitution als mehrstufig annimmt und bei der Überführung des abstrakten Bedeutungspotentials in eine konkrete Bedeutung (bei Cruse: *interpretation*) eine Unterscheidung vornimmt zwischen »pre-crystallization processes, processes preceding and leading up to crystallization, and post-crystallization processes« (siehe Croft/Cruse 2004, 100), wird die Bedeutungskonstitution in dieser Arbeit als einzelner, vor allem aber als einheitlicher Prozess modelliert, bei dessen Resultat nur zwischen einem *default construal* (der Grundbedeutung) und einem *full contextual construal* (der kontextualisierten Bedeutung) unterschieden wird.[60]

Wie können diese beiden Formen des *construal* nun auf Grundlage des WSM modelliert werden? Abb. 5.3 illustriert die Leitidee für die Beschreibung der Bedeutungskonstitution als einen Prozess der Transformation von Vektoren. Diese besteht darin, dass das durch einen Kookkurrenzvektor repräsentierte

60 Siehe dazu Abschnitt 3.2.2. Wie dort ausgeführt, nimmt Cruse selbst eine deutlich größere Differenzierung verschiedener Formen des *construals* vor (vgl. Anm. 35).

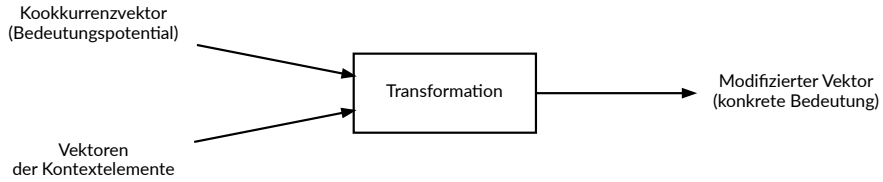


Abbildung 5.3: Ausgangspunkt der Bedeutungskonstitution ist ein Kookkurrenzvektor für das Zielwort, der zusammen mit den Kookkurrenzvektoren der Kontextwörter eingegeben wird. Im Prozess wird die ursprüngliche Repräsentation (das Bedeutungspotential) des Zielworts durch die Vektoren der Kontextelemente transformiert. Ergebnis der Transformation ist ein modifizierter Vektor, der als konkrete Bedeutung des Zielworts im betrachteten Kontext angesehen wird.

Bedeutungspotential in der konkreten Verwendung durch die mit in den Prozess eingebrachten Vektoren der Kontextelemente verändert wird.

Im Zuge der Transformation kommen gemäß Cruses Konzeption sogenannte *contextual constraints* zum Tragen, also Bedingungen, die durch den jeweiligen Kontext gestellt werden und die das *construal* steuern. Während Cruse hier zwischen rein sprachlichen und außersprachlichen Kontexten differenziert, beschränken sich die *contextual constraints* in der Modellierung auf den unmittelbaren linguistischen Kontext, das heißt auf genau die Kontextelemente, die zusammen mit dem Zielwort den Input für den Prozess bilden. Diese sind in der Modellierung ebenso wie das Zielwort durch Kookkurrenzvektoren repräsentiert und bringen damit gleichfalls ihr Bedeutungspotential ein – und somit auch ihre eigenen *conventionalized constraints*. In der Modellierung bestehen die *contextual constraints* demnach einfach in den *conventionalized constraints* der Kontextwörter; sie sind in Gestalt von Verwendungsmustern in deren Kookkurrenzvektoren implizit enthalten.

Auf Grundlage dieser Vorüberlegungen kann im Folgenden die hier beschriebene Idee einer Transformation auf das WSM übertragen werden. In Abschnitt 5.2.1 wird zunächst beschrieben, wie eine Veränderung der Repräsentation durch den eingegebenen Kontext mittels gängiger Vektoroperationen realisiert werden kann. Anschließend wird der Prozess um eine zusätzliche Gewichtung der Kontextelemente erweitert (Abschnitt 5.2.2). Ergebnis ist in beiden Fällen ein transformierter Vektor, der als konkrete Bedeutung im lokalen Kontext ausgelegt werden kann. In Abschnitt 5.2.3 wird beschrieben, wie die ermittelten Bedeutungen zueinander in Beziehung gesetzt werden können, um daraus ein erweitertes semantisches Profil zu erstellen.

5.2.1 Transformation durch den Kontext

Die Vorstellung einer Transformation bedeutet für die Modellierung im WSM, dass im Prozess die Repräsentation des Zielworts direkt verändert wird. Dies

geschieht unmittelbar durch den Einfluss der im Kontext auftretenden Wörter, und zwar dahingehend, dass im Ergebnis immer nur bestimmte Teile der enthaltenen Informationen betont werden, während andere in den Hintergrund treten. Dies lässt sich bei einer vektoriellen Repräsentation dadurch beschreiben, dass die Ausgangsrepräsentation (also der Kookkurrenzvektor eines Zielworts) durch eine Verschmelzung bzw. die Kombination mit den Repräsentationen der Kontextelemente (also durch deren Vektoren) modifiziert wird. Mit der Veränderung des Vektors ändert sich auch dessen Ausrichtung im Vektorraum. Er wird durch den Einfluss des Kontextes gewissermaßen in eine andere Richtung ›gezogen‹, schematisch dargestellt in Abb. 5.4. Die Veränderung der Repräsentation kann dabei als eine Art ›kontextuelle Aktivierung‹ angesehen werden, insofern durch die Veränderung der Repräsentation nur Teile des ursprünglichen Bedeutungspotentials aktiv sind.

Der hier verfolgte Ansatz orientiert sich im Wesentlichen an der Konzeption der *context vectors* nach Schütze (1992; 1998), wie sie in Abschnitt 4.1.3 beschrieben wurde. In Anlehnung an das dort beschriebene Vorgehen wird auf Grundlage der Vektoren der Kontextelemente ein neuer Vektor für den Kontext erstellt, der auf den Kookkurrenzinformationen sämtlicher enthaltener Wörter basiert – Schütze (1998) bezeichnet dies als Kookkurrenz zweiter Ordnung. Um möglichst nahe an der theoretischen Konzeption zu bleiben, wird dieser

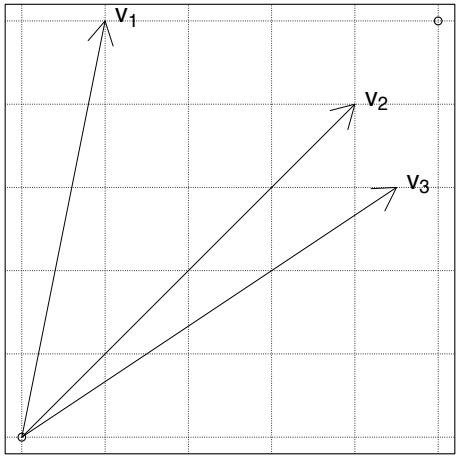


Abbildung 5.4: Kontextuelle Aktivierung im Vektorraum. Durch Kombination mit einem Kontextvektor V_1 wird die Richtung eines Zielwortvektors V_3 verändert. Resultat ist ein veränderter Vektor V_2. Die Transformation lässt sich als Aktivierung von bestimmten Teilen des Bedeutungspotentials interpretieren.

Kontextvektor – anders als bei Schütze – zunächst ohne das Zielwort erstellt und erst anschließend mit dessen Kookkurrenzvektor kombiniert.[61] Dadurch wird simuliert, dass das Bedeutungspotential des Zielworts im Sinne der »contextual constraints« gezielt durch den Kontext modifiziert wird. In dieser Konstellation ist der Prozess demnach zweischrittig: Zuerst werden die Kontextelemente zu einem einzelnen Kontextvektor zusammengefasst, indem über die Vektoren der Kontextelemente der Zentroid bzw. Schwerpunkt berechnet wird. Hierbei wird einfach für alle korrespondierenden Vektorelemente der jeweilige Mittelwert errechnet:[62]

$$\vec{\mu}(\omega) = \frac{1}{|\omega|} \sum_{\vec{x} \in \omega} \vec{x}$$

Im zweiten Schritt wird dieser Kontextvektor mit dem Ausgangsvektor (der das Bedeutungspotential des Zielworts repräsentiert) kombiniert, indem erneut der Zentroid berechnet wird. Im Sinne eines Prozesses geschieht beides ad hoc, also unmittelbar im Moment der Kontextualisierung. Die beiden Prozessschritte sind in Abb. 5.5 nochmals graphisch dargestellt.

Ergebnis der Transformation (bei Cruse: des *construal*) ist ein modifizierter Vektor, der die konkrete Bedeutung repräsentiert (bei Cruse: *interpretation*). Der Grad der Veränderung durch den Kontext richtet sich dabei nach der Stärke der in den beteiligten Vektoren enthaltenen Verwendungsmuster (bei Cruse: *constraints*): Sind die *conventionalized constraints* eines Wortes besonders stark ausgeprägt, so widerstehen sie denen des Kontextes und der ursprüngliche Vektor wird nur geringfügig geändert, so dass die Transformation in einer konventionalisierten Bedeutung bzw. einer Art Grundbedeutung resultiert (bei Cruse: *default construal*); sind sie eher schwach im Vergleich zu denen des Kontextes, werden sie von den *contextual constraints* (also den *conventionalized constraints* der Kontextelemente) überschrieben, und das Ergebnis der Transformation ist ein deutlich veränderter Vektor, der als die konkrete Bedeutung in dem aktuellen Kontext interpretiert werden kann (in Cruses Konzeption eine *fully construed meaning*). Diese Unterscheidung lässt sich anhand der in Abschnitt 3.2.2 aufgeführten Beispielsätze verdeutlichen, in dem die verschiedenen Kontextualisierungen des Verbs *spielen* verschiedene Interpretation ermöglichen:

61 Bei Schütze wird für den gesamten Kontext genau ein Vektor erstellt, indem der Schwerpunkt über die beteiligten Kookkurrenzvektoren gebildet wird – einschließlich des zu betrachtenden Wortes. Dadurch wird der Kontext gegenüber dem Zielwort deutlich stärker gewichtet, was der hier verfolgten Modellierung widersprechen würde: Hier muss der Ausgangsvektor eine größere Rolle spielen, um ein default construal zu ermöglichen.

62 Formel wiedergegeben nach Manning u.a. (2008, 360); die Berechnung des Schwerpunkts hat dabei zur Folge, dass der resultierende Vektor selbst keinem der bisherigen Datenpunkte entspricht – es entsteht somit tatsächlich ein neuer Vektor, der zuvor im Modell nicht enthalten war.

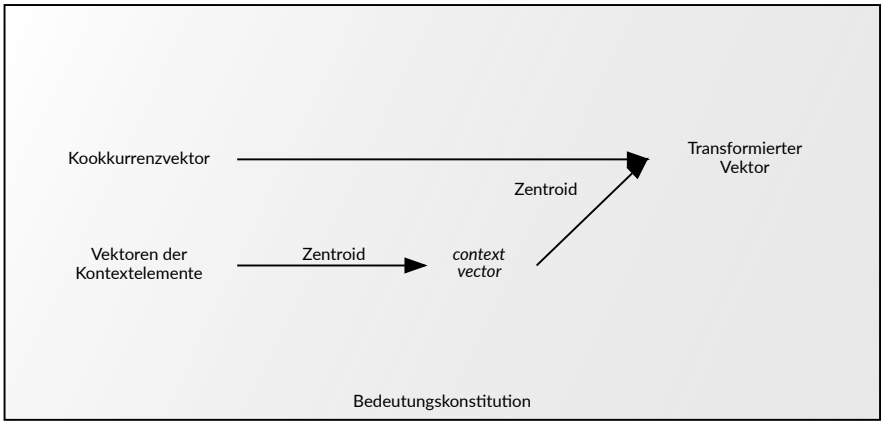


Abbildung 5.5: Modellierung der Bedeutungskonstitution als Transformation eines Kookkurrenzvektors durch den Kontext. Im ersten Schritt werden die Vektoren der Kontextelemente zu einem Kontextvektor zusammengefasst, indem deren Zentroid ermittelt wird; im zweiten Schritt wird der Kontextvektor durch eine erneute Berechnung des Zentroids mit dem Ausgangsvektor verschmolzen. Das Resultat ist ein neuer Vektor, der für die lokale Bedeutung steht.

Beispiel 5.1
a. Sie scheint gerne zu spielen.
b. Für sie scheint das keine Rolle zu spielen.
c. Sie scheint gerne Klavier zu spielen.
d. Sie spielt eben einfach gerne.

Die Vorhersage des Modells ist hier eine unterschiedliche Bedeutung für das Wort *spielen*, diese Bedeutungsvariation lässt sich im Modell dadurch erklären, dass das Wort *Rolle* (Beispiel 5.1.b) sowie *Klavier* (Beispiel 5.1.c) mit stärkeren *constraints* assoziiert sind und deshalb der Vektor für *spielen* durch sie stärker verändert als in Beispiel 5.1.a und Beispiel 5.1.d. Umgekehrt scheint in den gleichen Kontexten offenbar eine deutlich geringere Affinität zu den anderen Wörtern zu bestehen, sonst müsste in Beispiel 5.1.a und 5.1.c dem Wort *spielen* die gleiche Bedeutung zugeschrieben werden, da dort ja überwiegend die gleichen Wörter im Kontext auftreten. Es gibt demnach offenbar in vielen Kontexten Wörter, die einen stärkeren Einfluss ausüben als andere, was auf Unterschiede in den Verwendungsmustern zurückzuführen ist: enthält dieses Muster besonders einschlägige Verwendungsweisen (im Beispiel: »Rolle spielen« bzw. »Klavier spielen«), dann sind laut Modell dessen *conventionalized constraints* sehr prägnant, was sich auch im transformierten Vektor niederschlägt. Um diesen Einfluss im Sinne von kontextuellen Bedingungen noch deutlicher zu betonen, wird deshalb in einem zusätzlichen Schritt eine Gewichtung der Kontextelemente vorgenommen, der im Folgenden beschrieben wird.

5.2.2 Gewichtung der Kontexte

Wurden die *contextual constraints*, die im Zuge der Transformation zum Tragen kommen, bisher nur als Teil der Verwendungsmuster der Kontextvektoren in den Prozess eingebracht, werden diese *constraints* im Folgenden ganz explizit modelliert. Dies geschieht in Form einer Gewichtung auf Grundlage des lokalen Kontextes, welche die grundlegenden Affinitäten zwischen gemeinsam auftretenden Wörtern betont (und damit umgekehrt auch ihr Fehlen). Ziel der Gewichtung ist es, die in den Vektoren enthaltenen Verwendungseigenschaften (also ihre *conventionalized constraints*) stärker auf das betrachtete Zielwort zu beziehen und sie erst dann als contextual constraints zu interpretieren. Dadurch wird simuliert, inwiefern die im Kontext verfügbaren Informationen im Prozess der Bedeutungskonstitution genutzt werden, um zunächst die relevanten Teile des Kontextes zu identifizieren. Durch die Hinzunahme einer Gewichtung ergibt sich die erweiterte Prozessbeschreibung in Abb. 5.6.

Für den zusätzlichen Schritt der Gewichtung werden statistische Assoziationsmaße eingesetzt, wie sie sich in der statistischen Sprachverarbeitung für die Ermittlung von sogenannten »Kollokationen«[63] etabliert haben (siehe dazu Manning/Schütze 1999; Evert 2005). In der Sprachtheorie werden Kollokationen zumeist als regelhafte Wortverbindungen verstanden, wie zum Beispiel Mehrwortlexeme sie darstellen – auf Ebene einer statistischen Untersuchung bezeichnen sie dagegen schlicht das signifikant häufige gemeinsame Auftreten von Wortformen.[64] Nach Sinclair (1991, 10) sind Kollokationen als »distillation of the typical behaviour of a word« anzusehen: ähnlich den Kookkurrenzvektoren spiegeln Kollokationen die Verwendungseigenschaften von sprachlichen Ausdrücken direkt wider und können – in Anlehnung an das Zitat von Sinclair – damit als die eigentliche Essenz von Gebrauchskontexten angesehen werden.

63 Der Begriff der Kollokation wurde maßgeblich von John R. Firth im Rahmen seiner Contextual Theory of Meaning geprägt, in deren Mittelpunkt die zentrale Rolle des Kontextes für den Bedeutungsbegriff steht (siehe Firth 1957). Darin bezeichnen Kollokationen regelhafte oder typische Assoziationen zwischen sprachlichen Einheiten, deren gemeinsames Auftreten in erster Linie semantisch motiviert und deren gemeinsame Verwendung als normal anzusehen ist. Die Teilausdrücke müssen dabei nicht zwingend in einer festen Reihenfolge auftreten. So bilden beispielsweise *Hund* und *bellen* ebenso eine Kollokation wie *blond* und *Haar*. Nach Firth können aus den Kollokationen zu einem Wort dessen semantische Eigenschaften abgeleitet werden: »Collocations of a given word are statements of the habitual or customary places of that word« (siehe Firth 1957, 181). Ist die Assoziation zwischen den Teilausdrücken einer Kollokation besonders stark (wie z.B. in Redewendungen oder Mehrwortlexemen), dann tritt die Bedeutung der Teilausdrücke zu Gunsten einer gemeinsamen, kollokativen Bedeutung in den Hintergrund.

64 In vielen Bereichen der theoretischen Linguistik hat sich eine eher strenge Auslegung des Kollokationsbegriffs etabliert, nach der nur stark konventionalisierte Verwendungen wie idiomatische Ausdrücke und Mehrwortlexeme als Kollokation bezeichnet werden. Ein liberalerer Kollokationsbegriff in der Tradition von Firth hat sich v.a. in der vornehmlich britisch geprägten deskriptiven Lexikographie erhalten (siehe dazu u.a. Sinclair 1991; McEnery/Wilson 2001).

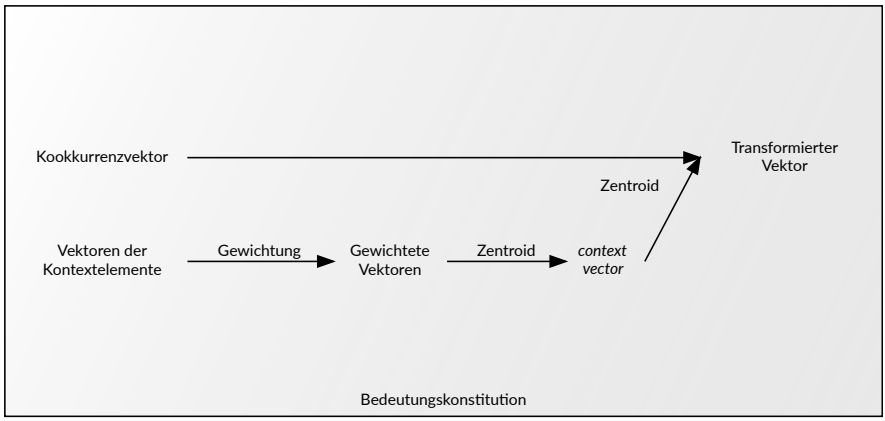


Abbildung 5.6: Erweiterte Prozessbeschreibung. Die Kontextelemente werden zunächst durch Anwendung von Assoziationsmaßen nach ihrer Affinität zum Zielwort gewichtet. Wie zuvor werden sie anschließend zu einem Kontextvektor zusammengefasst, der wiederum mit dem Ausgangsvektor verschmolzen wird.

Die Grenze zum Begriff der »Kookkurrenz«, der den Vektoren des WSM zugrunde liegt, ist dementsprechend fließend. So sind die für die Berechnung der Affinitäten benötigten Informationen direkt in den im Modell eingesetzten Kookkurrenzvektoren kodiert. Schon allein wenn man die Häufigkeit des gemeinsamen Vorkommens (also die Kookkurrenz) in Beziehung setzt zur jeweiligen Gesamthäufigkeit,[65] werden die grundlegenden Affinitäten sichtbar. Analog zu dieser recht einfachen Berechnung basieren auch komplexere Assoziationsmaße wie die Pointwise Mutual Information (PMI) nach Church/Hanks (1990) oder die Log-Likelihood-Ratio (LLR) nach Dunning (1993) letztlich auf dem Verhältnis des gemeinsamen Vorkommens gegenüber dem Auftreten in anderen Kontexten. PMI und LLR werden beide in dieser Arbeit für die Kontextgewichtung eingesetzt, deren Berechnung in Abschnitt 6.2.6 näher erläutert wird.

Ergebnis des Prozesses ist auch hier ein durch den Kontext transformierter Vektor, bei dem der Grad der Transformation von der Stärke der *contextual constraints* gegenüber den *conventionalized constraints* des Zielworts abhängt. Analog zum ungewichteten Vorgehen führen starke *conventionalized constraints* zu einem *default construal* (hier: zu einer geringen Veränderung des ursprünglichen Vektors), starke *contextual constraints* dagegen zu einem *full contextual construal* (hier: zu einer starken Anpassung der Ausgangsrepräsentation). Der Unterschied liegt in einem abweichenden Verständnis der *contextual constraints*: Anstatt die Repräsentation des Zielworts einfach durch sämtliche im Kontext auftretende

65 Dieses sehr einfache Maß wird u.a. von Sinclair (1991, 105f.) für die Identifikation von Kollokationen eingesetzt. Es ist zudem Bestandteil des im Information Retrieval häufig eingesetzten »tf.idf-Maßes« (vgl. dazu auch Abschnitt 6.2.5).

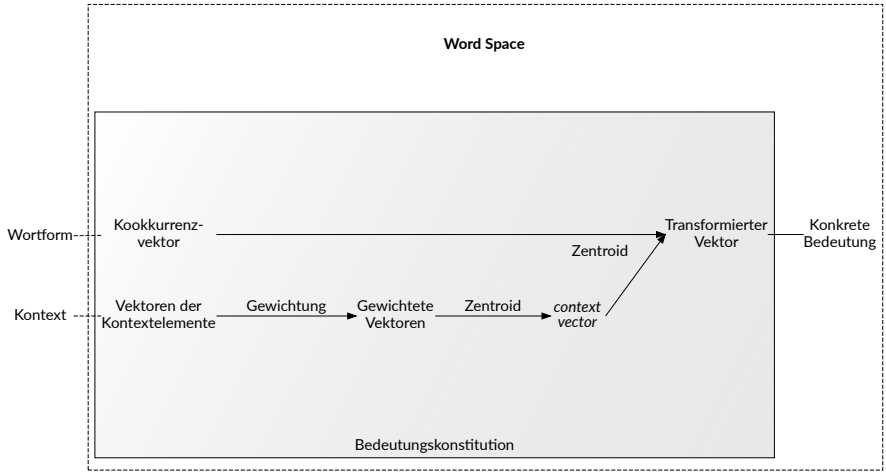


Abbildung 5.7: Bedeutungskonstitution im Wortraum. Wörter sind mit einem Kookkurrenzvektor assoziiert, der ihr Bedeutungspotential repräsentiert. Im Zuge der Kontextualisierung wird der Ausgangsvektor durch die Vektoren der Kontextelemente modifiziert, nachdem diese zunächst gewichtet und zu einem Kontextvektor zusammengefasst wurden. Ergebnis der Transformation ist ein neuer Vektor, der als lokale Bedeutung interpretiert werden kann.

Wörter gleichermaßen zu modifizieren, werden bei einer vorherigen Gewichtung all jene Wörter stärker berücksichtigt, die eine deutliche Affinität zum Zielwort aufweisen – oder, um es mit Sinclair auszudrücken: die Kontexte werden zunächst ›destilliert‹, um das typische Verhalten des Zielworts hervorzuheben.[66] In diesem Sinne werden die *contextual constraints* durch die Gewichtung hier auch sichtbar modelliert: Die *constraints* werden durch die Anwendung von Assoziationsmaßen ganz explizit ausgedrückt, während sie ohne Gewichtung nur implizit über die Verwendungsmuster in den Prozess einfließen. Input, Prozess und Output lassen sich nun zu dem in Abb. 5.7 dargestellten Gesamtbild der Bedeutungskonstitution im Vektorraum zusammenfassen.

Damit ist der Prozess der Bedeutungskonstitution im WSM im Grunde vollständig beschrieben: der Prozess wird als eine Transformation von Vektoren modelliert, die im Wesentlichen aus einer ad hoc (also im Zuge der Kontextualisierung) durchgeführten Kontextbewertung und der anschließenden Kombination von Kookkurrenzvektoren besteht. Aus dem Prozess resultiert ein transformierter Vektor, der nicht mehr das Bedeutungspotential repräsentiert, sondern – in

66 Dadurch soll gleichzeitig verhindert werden, dass Elemente mit starken *constraints* (also mit besonders prägnanten Verwendungsmustern), die nicht in Beziehung zum Zielwort stehen, einen zu großen Einfluss erhalten. Dies kann u.a. dann der Fall sein, wenn sehr lange Kontexte berücksichtigt werden, da hier die Wahrscheinlichkeit zunimmt, dass mehrere solcher ›starken‹ Wörter auftreten, so dass sich deren *constraints* gewissermaßen gegenseitig ›überschreiben‹ und damit aufheben.

Rückbezug auf den zugrunde gelegten *dynamic construal approach* nach Cruse – als *interpretation* ausgelegt werden kann, also als die lokale Bedeutung in einem konkreten Kontext.[67]

5.2.3 Mehrdeutigkeit im Vektorraum

Nachdem der Fokus der Beschreibung bisher auf der lokalen Bedeutungskonstitution in einzelnen Kontexten lag, soll es im Folgenden darum gehen, wie sich die verschiedenen *interpretations* (das heißt die lokalen Resultate des Prozesses) zueinander verhalten. Wie in Abschnitt 3.3 erörtert, markiert die Gesamtheit der möglichen *interpretations* in Cruses Konzeption einen Bereich im konzeptuellen Raum: »We can portray the total meaning potential of a word as a region in conceptual space, and each individual interpretation as a point therein« (siehe Croft/Cruse 2004, 109). Diese Vorstellung lässt sich unmittelbar im WSM modellieren, wobei zu beachten ist, dass die *interpretations* selbst nicht als Konzepte anzusehen sind, sondern nur als *contextually construed meanings*, also als jeweils nur lokal gültige Bedeutungen, die auf ein Konzept verweisen können (siehe dazu Abschnitt 3.2.1). Grundlage bildet eine mehrfache Kontextualisierung eines gleichen Wortes, wie sie in Abb. 5.8 schematisch dargestellt ist.

Im Modell ist jedes Wort mit einem Bedeutungspotential assoziiert, das durch einen Kookkurrenzvektor repräsentiert wird. Dieser bildet in den verschiedenen Kontextualisierungen den Ausgangspunkt für die jeweilige Bedeutungskonstitution, im Zuge derer der Vektor durch den Einfluss des Kontextes in einen neuen Vektor transformiert wird.[68]

Aufgrund der Unterschiede in den Kontexten weichen die aus dem Prozess resultierenden konkreten Bedeutungen bei jeder Kontextualisierung zumindest leicht voneinander ab. Analog zu Cruses Zitat bildet die Gesamtheit der *interpretations* damit auch in der Modellierung einen bestimmten Bereich im Vektorraum. Wie weit die einzelnen *interpretations* im Wortraum voneinander entfernt sind hängt davon ab, wie stark die jeweilige Transformation ausfällt: Ist das im Kookkurrenzvektor enthaltene Verwendungsmuster sehr heterogen, können die konkreten Bedeutungen unter Umständen stark voneinander abweichen; im umgekehrten Fall, also bei einem homogenen Verwendungsmuster, kommen dagegen die *conventionalized constraints* des Wortes stärker zum Tragen, so dass

67 Vgl. dazu auch Abschnitt 3.3.

68 Zwar ist in der zugrunde gelegten theoretischen Konzeption von Cruse eigentlich nicht vorgesehen, dass die Repräsentationen der konkreten Bedeutungen dauerhaft hinterlegt sind (im Modell muss im Grunde nur das Bedeutungspotential vorliegen, die konkreten Bedeutungen entstehen jeweils ad hoc im Zuge der Kontextualisierung und sind damit als temporäre Strukturen anzusehen, die nicht erhalten bleiben) – es ist im Modell jedoch ohne weiteres möglich, die lokalen Bedeutungen zu sammeln, um sie einer weitergehenden Analyse zu unterziehen.

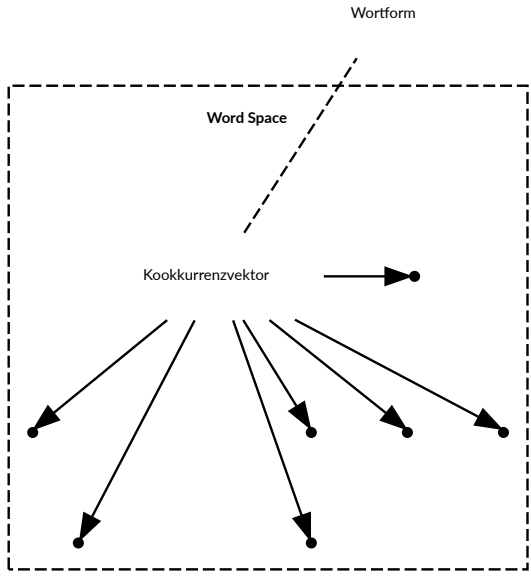


Abbildung 5.8: Schematische Darstellung der mehrfachen Kontextualisierung eines gleichen Wortes im Wortraum. Ausgehend von dem mit dem Wort assoziierten Kookkurrenzvektor konstituiert sich für jeden Kontext eine konkrete Bedeutung. Die Pfeile stehen für die entsprechenden Transformationen, deren Endpunkte für die resultierenden konkreten Bedeutungen.

die lokalen Bedeutungen sehr ähnlich zum Ausgangsvektor (und damit auch zueinander) sind.

Weil im Modell die Bedeutungsmöglichkeiten als implizit im Kookkurrenzvektor enthalten angenommen werden, kann die Gesamtheit der *interpretations* hier als eine explizite Darstellung des Bedeutungspotentials verstanden werden, ganz im Sinne des *total meaning potential* im Zitat von Cruse weiter oben. Aus dieser Ausdifferenzierung des Bedeutungspotentials ergibt sich für jedes Wort eine Art semantisches Profil, das die möglichen Bedeutungen eines Wortes umfasst.[69] Dieses Profil ist dabei in der Regel nicht gleichmäßig; vielmehr bilden die *interpretations* nach Croft/Cruse (2004, 109) mehr oder weniger einheitliche Gruppen: »the interpretations tend to cluster in groups showing different degrees of salience and cohesiveness, and between the groups there are relatively sparsely

69 Mit der Einschränkung, dass dies nur in Bezug auf das zugrunde gelegte Korpus gültig ist (insofern ein Korpus immer nur einen Ausschnitt der sprachlichen Möglichkeiten markiert), ist auch das erstellte semantische Profil nur als ein Ausschnitt anzusehen, der weder dauerhaft noch statisch ist.

inhabited regions.« Dank der numerischen Repräsentation durch Vektoren ist es in der Modellierung möglich, diese Gruppen durch die Anwendung gängiger Clustering-Algorithmen zu erschließen (siehe Abschnitt 6.2.7). Um das semantische Profil zu strukturieren, können ähnliche Bedeutungen mittels Clusteranalyse zusammengefasst und Teilbedeutungen voneinander abgegrenzt werden.[70]

5.3 Diskussion

Die in diesem Kapitel vorgeschlagene computerlinguistische Modellierung der Bedeutungskonstitution soll im Folgenden nochmals vor dem Hintergrund der in Abschnitt 3.2 getroffenen Annahmen bewertet werden. Zwar dient Cruses *dynamic construal approach* (Croft/Cruse 2004; Cruse 2004) weitgehend als konzeptionelle Vorlage; indem diese Arbeit sich auf den Prozess der Bedeutungskonstitution konzentriert, weicht die Modellierung jedoch in einigen wesentlichen Punkten von Cruses Gesamtkonzeption ab. So ist der *dynamic construal approach* in Cruses eigentlicher Konzeption noch deutlich detailreicher als hier dargestellt, jedoch werden diese Differenzierungen in der Modellierung auf das Wesentliche reduziert. So werden unter anderem einige der Konzepte ausgespart, auf die Cruse seinen Ansatz eigentlich gründet, etwa die Organisation konzeptueller Strukturen im Sinne der Frame-Semantik nach Fillmore (1976; 1982), siehe dazu Croft/Cruse (2004, Kapitel 2), oder die Strukturierung konzeptueller Kategorien in Anlehnung an die Prototypen-Theorie nach Rosch (1975; 1978), siehe dazu Croft/Cruse (2004, Kapitel 4).

Stattdessen basiert die Modellierung im Rahmen dieser Arbeit auf dem Word Space Model (WSM) nach Schütze (1992; 1993). Im WSM erfolgt zunächst die Repräsentation des Bedeutungspotentials eines Wortes über die algorithmische Erfassung seines Verwendungsmusters. Auf dieser Grundlage lässt sich unter Hinzunahme des Kontextes der Prozess der Bedeutungskonstitution durch gängige Vektoroperationen realisieren. Die Bedeutungskonstitution wird modelliert als eine Transformation der Ausgangsrepräsentation, die im Zuge der Kontextualisierung vollzogen wird. Beim Ergebnis des Prozesses wird in dieser Arbeit – anders als bei Cruse – nur eine Unterscheidung zwischen *default construal* und *full contextual construal* vorgenommen – also zwischen einer Grundbedeutung und einer konkreten Bedeutung im Kontext. Während das *default construal* bei jeder Kontextualisierung vollzogen wird, so dass immer zumindest eine Grundbedeutung vorliegt, greift das *full contextual construal* nur dann, wenn der Kontext ausreichende Informationen enthält, so dass das *default construal* gewissermaßen ›überschrieben‹ wird.

70 Solche Clusteranalysen sind dabei auch im Hinblick auf die Auswertung der Profile von großer Bedeutung, da sie in der Visualisierung eingesetzt werden können (siehe Abschnitt 6.2.8).

In Bezug auf die *contextual constraints*, die in Cruses Konzeption das *construal* regeln (bzw. in der Modellierung dann die Transformation), wird zudem eine Einschränkung auf rein sprachliche Kontexte vorgenommen. Diese ergibt sich unmittelbar aus der Verwendung des WSM, welches methodisch die Beschränkung auf geschriebene Sprache impliziert, wie sie in Korpora vorliegt. Die Beschränkung auf Korpusdaten steht dabei nur scheinbar in Widerspruch zum holistischen Anspruch der Kognitiven Linguistik. Zwar kann ein Korpus nicht die ganze Komplexität von Sprache erfassen, jedoch verfolgt die Kognitive Linguistik – anders als etwa strukturalistische Ansätze – auch gar nicht das Ziel, das Gesamtsystem einer Sprache aus Korpusdaten ableiten zu können.[71] Vielmehr akzeptiert die Kognitive Linguistik die vermeintliche ›Unvollständigkeit‹ des korpusbasierten Vorgehens: »[…] we do not attempt to account for all of language in every study. The usage-based model places variation, between groups and even between individuals, as an integral part of language« (siehe Glynn/Fischer 2010, 12). Nach Glynn legt der korpuslinguistische Zugang letztlich nur offen, dass es im Grunde unmöglich ist, das Phänomen Sprache als Ganzes zu betrachten. In dieser Perspektive können Korpora durchaus als hinreichend repräsentativ angesehen werden, da in der performanzorientierten Kognitiven Linguistik ohnehin immer nur Teile von Sprache untersucht werden.[72] Durch die Festlegung auf das WSM ist in dieser Arbeit somit nur eine Untersuchung sprachlicher Bedeutungen auf Grundlage von Sprache möglich. Zwar ergibt sich daraus, dass hier keine vollständige Modellierung von Cruses *dynamic construal approach* vorgenommen werden kann, da in diesem auch außersprachlichen Faktoren eine Rolle spielen; das ist jedoch auch nicht nötig: Unter der Annahme, dass die durch Cruse beschriebenen kognitiven Prozesse einen sprachlichen Widerhall finden, können mindestens die wesentlichen Aspekte im Wortraum modelliert werden.

Wenn man die Einschränkung auf Korpusdaten akzeptiert, die mit dem WSM verbunden ist, so ist es möglich, den Wortraum als ein Modell für den konzeptuellen Raum anzusehen: Weil in der Kognitiven Semantik auch semantische Strukturen als konzeptuelle Strukturen verstanden werden können (siehe Abschnitt 3.2.1), kann der semantische Raum in kognitiver Perspektive zumindest

71 Ziel des Strukturalismus ist nach Lyons (1971, 160) »[…] eine Technik oder ein Verfahren zu entwickeln, das auf ein Korpus von belegten Äußerungen angewendet werden könnte und das es […] erlauben würde, die Regeln der Grammatik mit Sicherheit aus dem Korpus selbst abzuleiten.«

72 Damit lässt sich nach Glynn/Fischer (2010) auch das Argument der ›negativen Evidenz‹ entkräften, das Chomsky (1959) gegen korpusbasierte Ansätze vorgebracht hat (vgl. dazu auch Anm. 20). Chomsky argumentierte, dass Korpora zwar korrekte Beispiele liefern können, jedoch keinerlei Aussage darüber zulassen, dass ein bestimmter sprachlicher Ausdruck nicht möglich ist – dies sei nur durch Introspektion und die Annahme eines idealisierten Sprecher-Hörers möglich. Dies gilt nach Glynn jedoch nur für regelbasierte Modelle wie das von Chomsky selbst, nicht aber für einen gebrauchsorientierten Ansatz wie die Kognitive Linguistik, in der Regeln als Generalisierungen über konkrete Verwendungen und damit als Epiphänomen angesehen werden (vgl. dazu Glynn/Fischer 2010, 13).

als eine Teilmenge des konzeptuellen Raums angesehen werden.[73] Dies ist jedoch nur dann möglich, wenn davon ausgegangen wird, dass im Wortraum nicht unmittelbar die Bedeutungen repräsentiert sind – zumindest nicht durch einfache Kookkurrenzvektoren. Stattdessen wurde in diesem Kapitel eine Umdeutung der Kookkurrenzvektoren vorgeschlagen: Da sie das gesamte Verwendungsmuster in einer einzigen Repräsentation vereinen, die im Sinne von Cruse nicht ausgedeutet ist, werden sie im Kontext dieser Arbeit als Bedeutungspotentiale angesehen. Sie beinhalten damit einerseits den *purport*, eine (unbestimmte) Menge an konzeptuellem Gehalt, andererseits die *conventionalized constraints*, welche in Gestalt von Verwendungsmustern die kombinatorischen Möglichkeiten festlegen. Auf dieser Grundlage können, durch Hinzunahme der *contextual constraints* (die in diesem Fall auf das direkte sprachliche Umfeld beschränkt sind), die eigentlichen *interpretations* erstellt werden.

Die Zielstruktur ist dann das, was Cruse als »full meaning potential« bezeichnet: eine »region in conceptual space« mit den »interpretations« als (nur temporär fixierte) Punkte darin (vergleiche Croft/Cruse 2004, 109),[74] die sich wiederum mittels Clusteranalyse in Gruppen organisieren lassen. Daraus lässt sich für jedes Wort eine Art semantisches Profil gewinnen. Anders als bei den Kookkurrenzvektoren, die die Bedeutungsmöglichkeiten nur implizit in Form eines Verwendungsmusters enthalten, sind die möglichen Bedeutungen in diesem Profil nunmehr explizit repräsentiert und können damit auch zueinander in Beziehung gesetzt werden.

Ob es tatsächlich ausreicht, das WSM umzudeuten bzw. die Repräsentationen einfach anders auszulegen, können im Grunde erst die konkreten Experimente beantworten: Wenn es möglich ist, das WSM dahingehend einzusetzen, dass sich die Bedeutungsvariation aus der Repräsentation ablesen lässt, dann spricht im Grunde nichts dagegen, aus dieser (sichtbar gemachten) Variation auf einen zugrunde liegenden (kognitiven) Prozess der Bedeutungskonstitution zu schließen. Wenn also auch die Repräsentation von Kontext zu Kontext variiert und dies mit sinnvoll interpretierbaren Veränderungen der Ähnlichkeit zu anderen Elementen einhergeht, dann kann dies als ein Indikator für die jeweils andere Konzeptualisierung der jeweiligen lokalen Bedeutung angesehen werden.

73 Der konzeptuelle Raum umfasst nach Auffassung der Kognitiven Semantik nicht nur linguistisches Wissen, sondern auch das (außersprachliche) Weltwissen (vgl. dazu Evans/Green 2006, 159).

74 Für das vollständige Zitat siehe Abschnitt 3.3.

6. Softwaretechnologische Umsetzung

Im Folgenden steht die softwaretechnologische Umsetzung der im vergangenen Kapitel beschriebenen Modellierung im Mittelpunkt. In der Unterscheidung verschiedener Beschreibungsebenen nach Marr (1982), die dem Aufbau dieser Arbeit zugrunde liegt (siehe Abschnitt 1.2), entspricht dies der dritten Ebene, auf der beschrieben wird, wie Algorithmus und Repräsentation konkret realisiert werden können. Die Modellierung der Bedeutungskonstitution, wie sie im vergangenen Kapitel skizziert wurde, erfolgt mittels eines mehrschrittigen Verfahrens. Die einzelnen Verfahrensbestandteile können dabei weitgehend als in sich geschlossene Teilaufgaben angesehen werden, zu denen es jeweils eine Reihe von Variationsmöglichkeiten gibt. Aus verfahrenstechnischer Sicht bietet sich hier der Einsatz eines komponentenorientierten Systems an, in dem die verschiedenen Verfahrensschritte in Software-Komponenten gekapselt werden können (siehe dazu Szyperski u.a. 2002). Für die softwaretechnologische Umsetzung des Verfahrens wird in dieser Arbeit deshalb das linguistische Komponentensystem Tesla[75] eingesetzt, das in der Sprachlichen Informationsverarbeitung an der Universität zu Köln entwickelt wurde (siehe vor allem Schwiebert 2012 sowie Hermes 2012).

Die Umsetzung in Tesla ist dabei als Ausdruck des in dieser Arbeit verfolgten methodischen Leitgedankens einer empirisch-experimentellen Herangehensweise an sprachwissenschaftliche Problemstellungen zu verstehen: In Tesla können die einzelnen Verfahrensschritte als separate Komponenten realisiert werden, die jeweils über eine Reihe von Parametern verfügen. Durch die Verknüpfung von Komponenten können verschiedene Experimente definiert werden, in denen die konkrete Ausführung des Verfahrens variiert wird; neben einer Variation der Parameter auf verschiedenen Verarbeitungsebenen umfasst dies auch die Variation der Verfahrensbestandteile selbst, indem diese in verschiedenen experimentellen Anordnungen ausgeführt werden. Von wesentlichem Interesse für diese Arbeit ist vor allem die konzeptuelle Nähe von Tesla zu den in der Einleitung getroffenen Aussagen über die Funktion von Experimenten und Simulationen: so ist ein experimentelles Vorgehen nach Rickheit u.a. (2010, 195f.) von entscheidender Bedeutung für den Erkenntnisgewinn in der Kognitiven Linguistik, insbesondere vor dem Hintergrund des von ihr propagierten gebrauchsorientierten Zugangs und der damit verbundenen empirischen Ausrichtung. Rickheit u.a. (2010) schreiben in diesem Zusammenhang:

> Die Vorteile von Simulationen […] sind
> (i) die Möglichkeit zur wiederholten Untersuchung eines Prozesses unter kontrollierten Bedingungen;
> (ii) systematische Variation und Kombination von Teilprozessen und
> (iii) Extrapolation in extreme Bereiche, bei denen Mensch und Tier gefährdet wären. (Rickheit u.a. 2010, 196)

75 Text Engineering Software Laboratory, siehe http://tesla.spinfo.uni-koeln.de (Zugriff vom 04.09.2017).

Während der letztgenannte Punkt im Zusammenhang mit computerlinguistischer Forschung vermutlich eher von untergeordnetem Interesse ist (siehe dazu auch Schwiebert 2012, 77f.), lassen sich aus den beiden ersten Punkten die Möglichkeit zur Kontrolle, Wiederholbarkeit und Variation als allgemeine Anforderungen an wissenschaftliche Experimente ableiten, die auch für sprachwissenschaftliche Experimente gelten (vergleiche Rickheit u.a. 2010, 196). Aus Perspektive der Kognitiven Lingusitik übernimmt Tesla in dieser Arbeit somit die Funktion als »virtuelles Labor, in dem virtuelle Experimente durchgeführt werden« (siehe Rickheit u.a. 2010, 196).

Im Folgenden werden zunächst die zentralen Konzepte von Tesla erläutert (Abschnitt 6.1). Der Fokus liegt dabei auf den für diese Arbeit relevanten funktionalen Aspekten; für zusätzliche technische und konzeptuelle Details sei vor allem auf die Arbeit von Schwiebert (2012) verwiesen.[76] Anschließend werden die für die Modellierung vorgesehenen Verfahrensschritte sowie die dabei eingesetzten Komponenten beschrieben (Abschnitt 6.2). Im Sinne des Laborgedankens können die für diese Arbeit erstellten Komponenten als eine spezialisierte Laborausstattung angesehen werden, die auch über den konkreten Anwendungsfall hinaus eingesetzt werden kann. Dies wird in der abschließenden Zusammenfassung thematisiert (Abschnitt 6.3), in der die für diese Arbeit zentralen Konzepte von Tesla noch einmal kurz zusammengefasst werden.

6.1 Das Text Engineering Software Laboratory (Tesla)

Leitgedanke bei der Konzeption von Tesla war die Umsetzung einer virtuellen Arbeitsumgebung für empirisch-experimentelle Forschung in Computer- und Korpuslinguistik (siehe Schwiebert 2012). Rein funktional betrachtet ist Tesla dabei zunächst ein linguistisches Komponentensystem, das eine integrierte Umgebung für die Entwicklung und Ausführung von Software-Komponenten zur Verarbeitung textueller Daten bietet. Die Grundidee linguistischer Komponentensysteme besteht in der Kapselung von Verarbeitungsschritten in Software-Komponenten, die über wohldefinierte Schnittstellen Daten austauschen und in Abhängigkeit verschiedener Anwendungsszenarien zu komplexen Verarbeitungsketten zusammengefügt werden können. Aufgabe der einzelnen Komponenten ist die schrittweise Anreicherung textueller Daten mit zusätzlichen Informationen – in diesem Zusammenhang in der Regel als »Annotationen« bezeichnet – die aus der Analyse der Daten gewonnen werden und im Zuge derer auch externe Quellen wie Lexika, Wissensbasen oder ähnliche zum Einsatz kommen können. In seiner Grundkonzeption ist Tesla damit vergleichbar mit anderen komponentenorientierten

76 Zu den in Tesla umgesetzten konzeptuellen Grundlagen siehe auch Hermes (2012).

Systemen zur Verarbeitung textueller Daten wie zum Beispiel GATE[77] und UIMA[78], wobei Tesla in Abgrenzung zu diesen Systemen einige Besonderheiten aufweist, die speziell im Kontext dieser Arbeit relevant sind.[79]

Maßgeblich ist insbesondere die namensgebende Labormetapher sowie der damit verbundene Experimentbegriff, nach dem sich auch die Herangehensweise in der Arbeit mit Tesla richtet; diese werden in Abschnitt 6.1.1 erläutert. Darauf aufbauend beschreibt Abschnitt 6.1.2, wie die mit der Labormetapher verbundenen Konzepte in Tesla umgesetzt sind: so bietet Tesla einerseits eine Umgebung für die Entwicklung von Komponenten, andererseits aber auch einen graphischen Editor für die Konfiguration von Experimenten, der als die eigentliche Umsetzung der Labormetapher angesehen werden kann. Von Bedeutung für diese Arbeit sind zudem die weitreichenden Freiheiten bei der Entwicklung von Komponenten, die sich unter anderem aus der Orientierung an den Möglichkeiten der zugrunde gelegten Programmiersprache Java ergeben, da dies den in dieser Arbeit verfolgten empirisch-experimentellen Ansatz zusätzlich begünstigt. Das hierfür wesentliche Komponentenmodell von Tesla wird in Abschnitt 6.1.3 beschrieben.

6.1.1 Experimente im virtuellen Labor

Die zentralen Konzepte von Tesla basieren auf der auch im Namen enthaltenen Vorstellung eines virtuellen Labors, in dem sprachwissenschaftliche Experimente definiert und durchgeführt werden können.[80] In seiner Konzeption orientiert sich Tesla dabei am Aufbau eines naturwissenschaftlichen Labors, jedoch sind sowohl das Labor als auch dessen Ausstattung hier virtuell: Anstelle von Substanzen, Kolben und Reagenzgläsern besteht die Ausstattung im Wesentlichen aus Daten bzw. Datenquellen, Algorithmen und Datenstrukturen. Die Analogie zum naturwissenschaftlichen Labor besteht vor allem in der Art und Weise, wie hier Forschung betrieben wird. Von entscheidender Bedeutung ist in diesem Zusammenhang der Begriff des Experiments: Im sprachwissenschaftlichen Erkenntnisprozess haben Experimente die Aufgabe, empirische Daten zu gewinnen, um Hypothesen zu

77 GATE (General Architecture for Text Engineering) wurde von Hamish Cunningham als Referenzimplementation des von ihm in Cunningham (2000) eingeführten Konzepts der Software Architecture for Language Engineering (SALE) umgesetzt.

78 UIMA (Unstructured Information Management Architecture) wurde ursprünglich von IBM Research entwickelt (vgl. Ferrucci/Lally 2003; 2004), ist jedoch bereits seit 2005 Open Source verfügbar und wird mittlerweile von der Apache Software Foundation betreut (siehe http://uima.apache.org; Zugriff vom 21.02.2018) und kontinuierlich weiterentwickelt.

79 Für einen ausführlichen Vergleich der genannten Systeme hinsichtlich ihrer Gemeinsamkeiten und Unterschiede, auch in Bezug auf Tesla, sei hier auf Schwiebert (2012, Kapitel 3) verwiesen.

80 Tesla ist genau genommen nicht auf sprachwissenschaftliche Fragestellungen bzw. allgemeiner auf die Verarbeitung sprachlicher Daten beschränkt. Vielmehr können in Tesla grundsätzlich alle Arten von textuellen Daten (im Sinne von sequentiell gefassten Zeichenketten) verarbeitet werden. Bezüglich eines entsprechend erweiterten Textbegriffs sei auf Hermes (2012) verwiesen.

überprüfen, etwa in Bezug auf die Eigenschaften der untersuchten Daten oder bezüglich der Auswirkung bestimmter Parameter in der Verarbeitung der Daten. Die Modellierung von sprachlichen Prozessen in Form von Experimenten ist dementsprechend auch eines der zentralen Konzepte von Tesla. Experimente werden in Tesla als Komponenten-Workflows repräsentiert, in denen sämtliche Verfahrensbestandteile unter Berücksichtigung ihrer gegenseitigen Abhängigkeiten abgebildet sind, exemplarisch dargestellt in Abb. 6.1.

Aus technischer Sicht wird die Experimentdefinition als XML-Datei hinterlegt, in der der vollständige Versuchsaufbau spezifiziert ist, einschließlich der verwendeten Datenquellen, der eingesetzten Komponenten sowie ihrer jeweiligen Parameter, über die die Art der Ausführung spezifiziert wird. Bei der Verarbeitung werden die einzelnen Komponenten separat ausgeführt, sobald die von ihnen benötigten Daten zur Verfügung stehen; sofern keine gegenseitigen Abhängigkeiten aufgelöst werden müssen, kann dies auch parallel geschehen, da die Verarbeitung der Komponenten in gesonderten Threads erfolgt.

Ein essentieller Bestandteil experimenteller Forschung ist die umfassende Dokumentation. Bei der Ausführung der Komponenten werden alle Zwischenergebnisse, das heißt sämtliche von einer Komponente produzierten Daten, zusammen mit der Experimentkonfiguration im Sinne eines virtuellen Laborhefts gespeichert. Für die Speicherung nutzt Tesla einen Annotationsgraphen (siehe Bird/Liberman 2001), auf den in den verschiedenen Verarbeitungsstufen zugegriffen werden kann.[81] Die Ausgangsdaten bleiben dabei stets unverändert, wodurch eine strikte Trennung der Daten von ihrer Interpretation gewährleistet ist, die als eine der zentralen wissenschaftstheoretischen Forderungen in der Korpuslinguistik gilt (vergleiche dazu etwa McEnery/Wilson 2001). Durch die Möglichkeit des Zugriffs auf Teilergebnisse können die Komponenten zudem einzeln analysiert werden, wodurch auch der Ausgang des gesamten Experiments besser nachvollzogen werden kann.

6.1.2 Arbeiten im virtuellen Labor

Technisch setzt Tesla eine typische Client-Server-Architektur um: Auf Clientseite können Komponenten entwickelt und in Form von Experimenten definiert werden, die anschließend serverseitig verarbeitet werden. Dies ist insbesondere bei sehr rechenintensiven Operationen von Vorteil, da die Verarbeitung auf einen leistungsfähigen Server ausgelagert werden kann, während die Systemressourcen

81 Der Annotationsgraph ist hier als abstraktes Konzept zu verstehen (vgl. Schwiebert 2012, 143f.), tatsächlich erfolgt die Speicherung unter Nutzung verschiedener Persistenzframeworks in mehreren Datenbanken. Für Details zur Auswahl und Begründung der in Tesla eingesetzten Persistenzmechanismen siehe Schwiebert (2012, Kapitel 4.2).

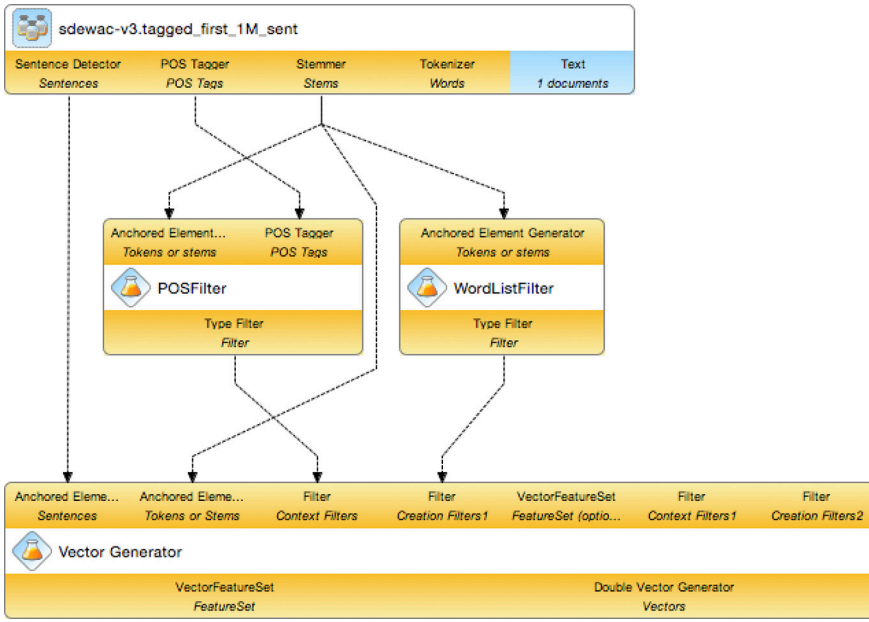


Abbildung 6.1: Beispiel eines Tesla-Experiments (Screenshot des graphischen Editors). Im hier dargestellten Workflow werden Wortvektoren auf Grundlage einer Teilmenge des SdeWaC-Korpus erstellt. Der WordListFilter definiert, für welche Wörter (bzw. hier: Stems) Vektoren erstellt werden, der POSFilter wiederum legt fest, welche Wortart (Part Of Speech, POS) die Kookkurrenten haben müssen.

auf Clientseite frei bleiben.[82] Der TeslaClient basiert auf dem Eclipse Framework[83] unter Nutzung des Plugin-Konzepts der Rich Client Platform.[84] Auf Clientseite stehen zwei Anwendungskontexte zur Verfügung, die auf Grundlage des Eclipse-Frameworks als eigene Perspektiven realisiert wurden, in denen unterschiedliche Schwerpunkte gesetzt werden: während in der Developer Perspective die Entwicklung von Komponenten im Mittelpunkt steht, dient die Linguist Perspective vor allem der Konfiguration und Ausführung von Experimenten.

In der Developer Perspective steht dem Entwickler eine vollwertige Java-IDE (Integrated Development Environment) zur Verfügung. Um die Entwicklung neuer Komponenten zu erleichtern, wurde die Eclipse-eigene IDE um einige Tesla-spezifische Menüpunkte erweitert. So steht unter anderem ein Wizard zur

82 Die im Client definierten Experimente werden als leichtgewichtige XML-Dateien an den Server gesendet und dort ausgeführt. Der Server wurde als eigenständige Anwendung auf Basis des Spring Framework (siehe https://spring.io – Zugriff vom 21.02.2018) umgesetzt. Zur Architektur des Tesla-Servers sowie zu technischen Details der Implementation siehe Schwiebert (2012, Kapitel 4.3).

83 Siehe https://eclipse.org (Zugriff vom 21.02.2018).

84 Siehe dazu Schwiebert (2012, Kapitel 4.1.7.1).

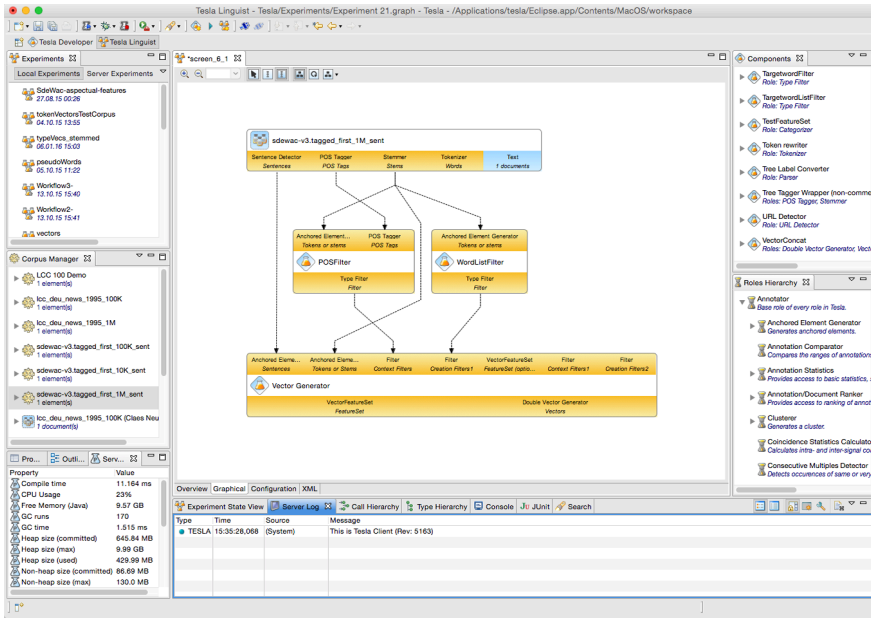


Abbildung 6.2: Die Linguist Perspective von Tesla (Screenshot). Rund um den zentralen graphischen Editor sind verschiedene Views arrangiert, über die unter anderem auf die Experimentdefinitionen sowie auf vorhandene Datenquellen und Komponenten zugegriffen werden kann.

Erstellung von Komponenten zur Verfügung, zudem wurden zusätzliche Ansichten hinzugefügt, die eine Übersicht der vorhandenen Komponenten sowie der verfügbaren Rollen geben.[85] Des Weiteren wurde ein lokaler Tesla-Server integriert, der direkt aus dem Client gestartet werden kann. Beim Start werden die neu entwickelten Komponenten auf den Server übertragen, so dass sie direkt getestet werden können. Für die Erstellung und Ausführung von Experimenten steht mit der Linguist Perspective eine eigene Umgebung bereit, die als die eigentliche Umsetzung des virtuellen Labors angesehen werden kann. Die Linguist Perspective umfasst verschiedene Ansichten, in denen unter anderem die lokal vorhandenen sowie die serverseitig bereits ausgeführten Experimente verwaltet und evaluiert werden können. In weiteren Ansichten sind zum einen die vorhandenen Komponenten und Rollen, zum anderen die verfügbaren Korpora und Datenquellen aufgelistet. Diese können per Drag-and-drop in einen graphischen Experiment-Editor gezogen werden, in dem die Experimente in Form von Komponenten-Workflows (siehe Abb. 6.2) arrangiert werden können.

Der in der Linguist Perspective integrierte graphische Editor stellt ein Alleinstellungsmerkmal dar gegenüber anderen komponentenorientierten Systemen wie den

85 Zum Begriff der Rolle im Zusammenhang mit Tesla siehe Abschnitt 6.1.3.

oben genannten UIMA oder GATE. Der Editor bietet eine zusätzliche Abstraktionsschicht für den Anwender, in der die Komponenten, die in einem Versuchsaufbau eingesetzt werden, auf ihre Funktionalität reduziert werden. Der Editor sorgt damit für eine geringere Komplexität in den Anforderungen an den Benutzer, so dass dieser auch ohne detaillierte Kenntnisse bezüglich der konkreten Implementation einzelner Verfahrensbestandteile bzw. Komponenten das System benutzen kann.

Neben dem Experiment-Editor stellt Tesla zusätzliche Ansichten bereit, beispielsweise um die Parametrisierung der Komponenten zu konfigurieren und um auf die Ergebnisse zuzugreifen. Für die Auswertung der Experimente stehen verschiedene Formen der Ergebnisdarstellung zur Verfügung. Neben einer Übersicht

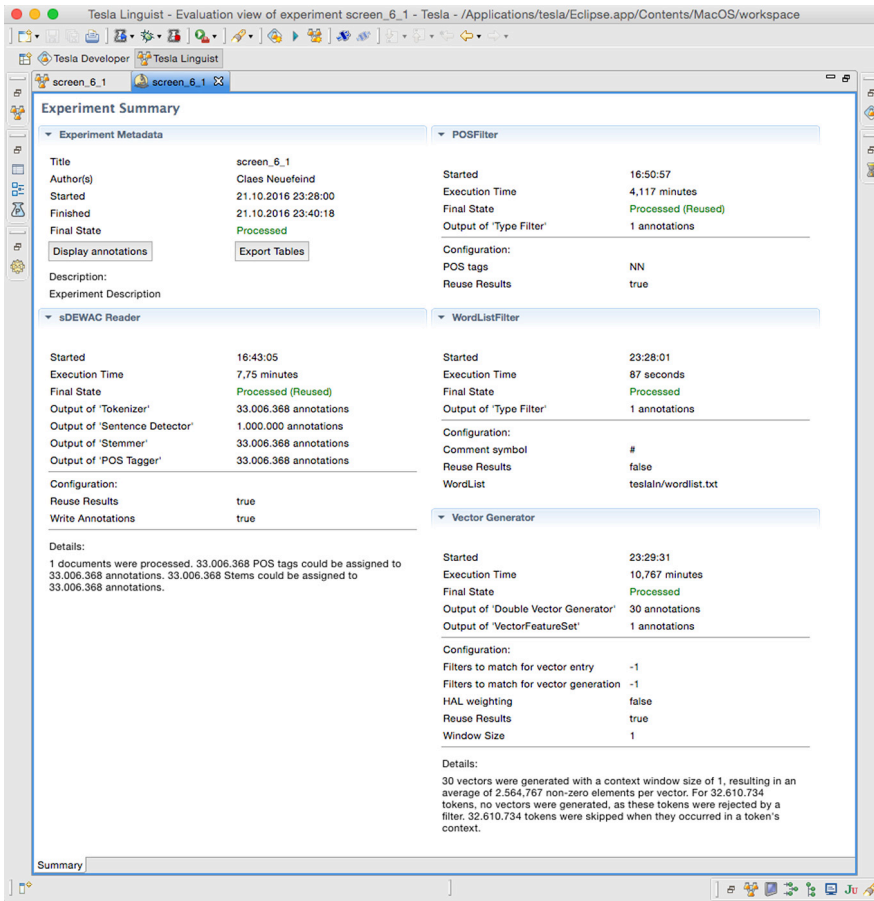


Abbildung 6.3: Ergebnisdarstellung in Tesla (Screenshot). In der Evaluation View werden die Ergebnisse eines ausgeführten Experiments zusammengefasst, unterteilt nach Komponenten. Die im Experiment erstellten Annotationen und Tabellen können über die entsprechenden Schaltflächen (links oben) exportiert werden.

mit allgemeinen Informationen zu den einzelnen Komponenten in einer eigenen Ansicht (siehe Abb. 6.3) besteht die Möglichkeit, gezielt die Ergebnisse einzelner Komponenten in Form von CSV- oder LaTeX- Tabellen zu exportieren, etwa um sie in externen Programmen weiterzuverarbeiten oder um sie direkt in einer Veröffentlichung einzubinden (wie beispielsweise in dieser Arbeit geschehen).

Beim Ergebnisexport kann festgelegt werden, zu welchen Komponenten die Ergebnisse visualisiert werden sollen und welche Zugriffsmethoden der Komponente dabei zu berücksichtigen sind. Da die Evaluation oftmals experimentspezifischen Anforderungen unterliegt, kann sie auch durch entsprechend spezialisierte Komponenten realisiert werden, etwa durch spezifische Formen der Visualisierung (siehe dazu auch Abschnitt 6.2.8) oder indem ein sogenannter »Goldstandard« zum Vergleich herangezogen wird.

6.1.3 Das Tesla Role System

Wie oben beschrieben ist Tesla in Bezug auf die konkrete Ausführung der Experimente als ein linguistisches Komponentensystem anzusehen, das auf die schrittweise Verarbeitung und Anreicherung textueller Daten ausgelegt ist. Eine wesentliche Anforderung an Komponentensysteme ist ein Komponentenmodell mit wohldefinierten Schnittstellen, die die Weitergabe der zu verarbeitenden Daten sowie der Annotationen zwischen den einzelnen Komponenten regeln. Grundlage hierfür ist eine Typisierung der verarbeitenden Komponenten. Diese wird in Tesla durch das Tesla Role System (TRS) umgesetzt (siehe Hermes/Schwiebert 2010), in dem festgehalten ist, welche Funktion eine Komponente in der Verarbeitung einnimmt. Das TRS basiert auf dem Konzept linguistischer Rollen, anhand derer die Ein- und Ausgabeschnittstellen von Komponenten spezifiziert werden. Rollen definieren dabei nicht nur die Art der Annotationen, die von einer Komponente produziert oder konsumiert werden, sondern legen gleichzeitig auch die für diese Annotationen vorgesehenen Zugriffsmöglichkeiten fest.

Im Unterschied zu den oben genannten Komponentenframeworks, bei denen die Typisierung in der Regel anhand der von der Komponente produzierten Annotationen vorgenommen wird, erfolgt die Typisierung im TRS danach, welche Rolle(n) eine Komponente in der Verarbeitung erfüllt. Mit dem TRS wird somit eine Abstraktion über konkrete Datenstrukturen vorgenommen: so können die Rollen in der konkreten Realisierung durch Komponenten auf verschiedene Art erfüllt werden, was eine hohe Flexibilität bei der Umsetzung verschiedener funktionaler Rollen erlaubt. Ein Beispiel hierfür ist bereits auf der einfachsten Ebene der Verarbeitung zu finden: so steht für das initiale Einlesen der zu verarbeitenden Daten eine Vielzahl verschiedener Reader-Komponenten bereit (siehe dazu auch Abschnitt 6.2.1), die aus verarbeitungstechnischer Sicht die gleiche Funktion übernehmen. Während sie somit funktional die gleiche Rolle erfüllen, kann die konkrete Realisierung durch die verschiedenen Reader-Komponenten sehr

unterschiedlich ausfallen, je nachdem, in welchem Format die Datenquelle vorliegt und welche Zugriffe die Komponente auf die Daten gestatten soll. Da die Rollen mit den Ein- und Ausgabeschnittstellen der Komponenten assoziiert sind, können die verschiedenen Komponenten unter Umständen auch mehr als eine Rolle implementieren, etwa wenn sie verschiedene Arten von Annotationen wie zum Beispiel Token, Lemma oder Wortart weitergeben sollen.

Die softwaretechnologische Realisierung des Rollenkonzepts basiert ganz wesentlich auf dem durch die zugrunde gelegte Programmiersprache Java realisierten Paradigma der Objektorientierung. So werden Komponenten in Tesla als Java-Klassen realisiert, die mittels des TRS über objektorientierte Schnittstellen auf Basis von Java-Interfaces verfügen. Das TRS verfolgt dabei einen API-ähnlichen Ansatz: die Rollendefinition besteht aus zwei Java-Interfaces, in denen die grundlegende Funktionalität hinsichtlich der Art der Annotation und der zugehörigen Zugriffsmöglichkeiten festgelegt ist. Die tatsächliche Implementation der Interfaces kann in der konkreten Realisierung durch Komponenten auf unterschiedliche Art und Weise erfolgen, schematisch dargestellt in Abb. 6.4.

Da die Interfaces erst in der konkreten Umsetzung durch eine Komponente ausprogrammiert werden, können gleiche Rollen durch unterschiedliche

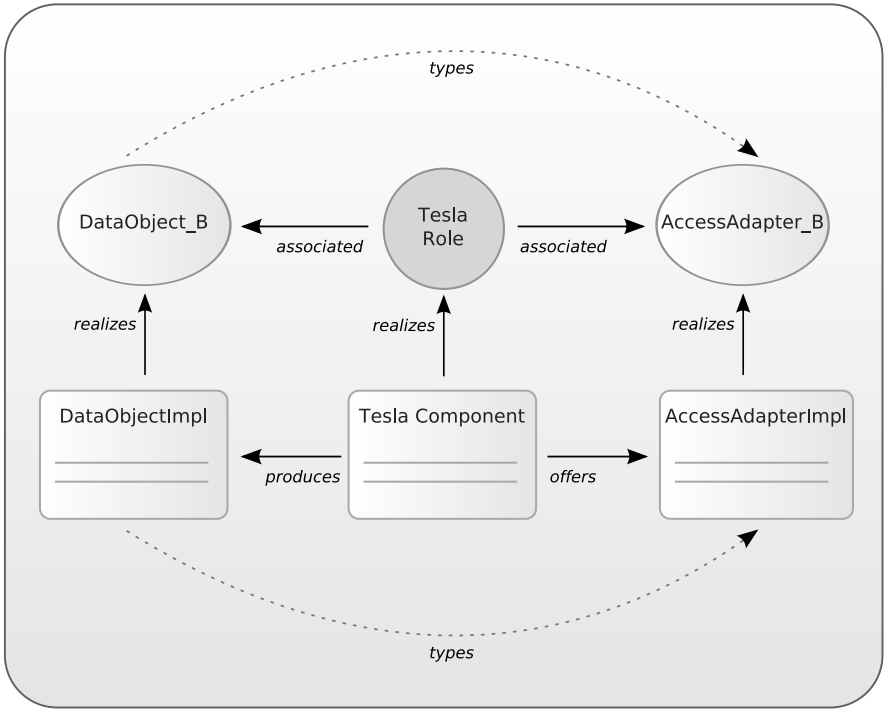


Abbildung 6.4: Schematische Darstellung des Tesla Role System (Grafik übernommen aus Hermes/Schwiebert 2010). Eine Rolle besteht aus einem AccessAdapter- und einem DataObject-Interface, die in der konkreten Realisierung durch eine Komponente implementiert werden.

Implementationen realisiert werden, ohne dass systemseitig vorgegebene Datenstrukturen beachtet werden müssen. Eine Rolle kann somit durch verschiedene, beliebig komplexe Komponenten erfüllt werden, wodurch das TRS weitreichende Freiheiten bei der Entwicklung von Komponenten gestattet, wobei gleichzeitig das für Komponentensysteme konstitutive Prinzip der funktionalen Austauschbarkeit umgesetzt wird. Für den Einsatz von Tesla für die in dieser Arbeit beschriebene Modellierung der Bedeutungskonstitution auf Grundlage des Word Space Model ist eine Reihe von Anpassungen nötig. Die hierfür im Zuge der Arbeit umgesetzten zusätzlichen Funktionen und Komponenten werden im Folgenden beschrieben.

6.2 Verfahrensschritte und Komponenten

Für die Umsetzung des in Kapitel 5 beschriebenen Modells der Bedeutungskonstitution wird eine Reihe von Komponenten benötigt, die nach dem Prinzip der Variierbarkeit auf verschiedenen Ebenen austauschbar sind und über ihre Parametrisierung verschiedene Konfigurationen ermöglichen. Die im Folgenden beschriebenen Komponenten dienen hier als Grundbausteine für die Umsetzung des Modells im Rahmen verschiedener Experimente.[86] Die konkreten Workflows sowie die zugehörigen Parametrisierungen werden im Kontext der jeweiligen Experimente beschrieben. Im Wesentlichen lassen sich folgende grundlegende Verarbeitungsstufen unterscheiden:

- Daten einlesen (Abschnitt 6.2.1)
- Vorverarbeitung (Abschnitt 6.2.2)
- Erstellung von Kookkurrenzvektoren (Abschnitt 6.2.3)
- Normalisierung der Vektoren (Abschnitt 6.2.4)
- Gewichtung der Vektoren (Abschnitt 6.2.5)
- Repräsentation von Einzelvorkommen (Abschnitt 6.2.6)
- Clusteranalyse (Abschnitt 6.2.7)
- Visualisierung (Abschnitt 6.2.8)

Zusätzlich wird in Abschnitt 6.2.9 die Auswahl geeigneter Beispielwörter beschrieben, die in den Experimenten untersucht werden sollen. Die einzelnen Verarbeitungsschritte sowie die dabei benötigten Komponenten werden im Folgenden

86 Nicht alle der aufgeführten Komponenten werden in den konkreten Experimenten tatsächlich eingesetzt. Da jedoch eines der Ziele dieser Arbeit in der Bereitstellung einer Arbeitsumgebung für distributionell motivierte Untersuchungen besteht, wurden auf allen Ebenen auch zusätzliche Alternativen integriert. Darüber hinaus steht in Tesla eine größere Anzahl weiterer Komponenten aus anderen Anwendungskontexten bereit (vgl. dazu Schwiebert 2012; Hermes 2012), von denen die meisten jedoch für diese Arbeit nicht unmittelbar relevant sind.

jeweils nur kurz skizziert; ausführlichere Beschreibungen der einzelnen Komponenten finden sich in Anhang A.[87]

6.2.1 Korpora

Für die Durchführung des Vorhabens bedarf es zunächst einer geeigneten Datenbasis in Form von großen Korpora. Im Rahmen dieser Arbeit werden zum einen die Korpora der Leipzig Corpora Collection (LCC, siehe Quasthoff u.a. 2006; Goldhahn u.a. 2012) eingesetzt, die von der Universität Leipzig bereitgestellt werden.[88] Die LCC umfasst Korpora in einer Vielzahl verschiedener Sprachen unter Einbeziehung vergleichbarer Ressourcen (zum einen Zeitungen und Pressedienste, zum anderen aus dem Internet bezogene Texte). Die Texte sind in einzelne Sätze zerlegt, welche in zufälliger Folge als Plain Text und als MySQL-Datenbanken zur Verfügung stehen. Unvollständige Sätze und fremdsprachliches Material wurden entfernt. Die LCC stellt die Korpora in Größen ab 10.000 bis zu 3 Millionen Sätzen bereit; in dieser Arbeit werden zwei deutschsprachige Korpora im Umfang von jeweils 1 Million Sätzen verwendet. Zum anderen wurde als Alternative zu den LCC-Korpora das über die WaCky-Initiative[89] bereitgestellte SdeWaC-Korpus eingebunden. SdeWac bezeichnet eine bereinigte Teilmenge des deutschen WaCky-Webkorpus (das »Stuttgart deWaC«, siehe Baroni/Kilgariff 2006), in der Satz-Duplikate und fremdsprachliches Material entfernt wurden. Anders als bei den Korpora der LCC wurden die Sätze im SdeWaC-Korpus bereits linguistisch vorverarbeitet, indem sie mit dem Tokenizer von Schmid (2000) in Tokens eingeteilt und anschließend mit dem TreeTagger (Schmid 1994) lemmatisiert und unter Verwendung des Stuttgart-Tübingen-TagSet (STTS)[90] mit Wortarten ausgezeichnet wurden. Das SdeWaC-Korpus enthält ca. 44 Millionen Sätze mit insgesamt über 846 Millionen Tokens, wobei im Rahmen dieser Arbeit nur eine auf die ersten 1 Million Sätze beschränkte Teilmenge verwendet wurde, um die Vergleichbarkeit zu den LCC-Korpora zu wahren. Die Einbindung der Korpora in Tesla erfolgt über die folgenden spezialisierten Reader-Komponenten, die den Zugriff auf die Korpusdaten regeln:

- LCC Reader
- SdeWac Reader

87 Siehe dazu auch http://tesla.spinfo.uni-koeln.de (Zugriff vom 04.09.2017).

88 Siehe http://corpora2.informatik.uni-leipzig.de/download.html (Zugriff vom 21.02.2018).

89 »The Web-As-Corpus Kool Yinitiative«, siehe Baroni u.a. (2009) bzw. http://wacky.sslmit.unibo.it (Zugriff vom 21.02.2018).

90 Siehe http://www.ims.uni-stuttgart.de/forschung/ressourcen/lexika/TagSets/stts-table.html (Zugriff vom 21.02.2018).

Die LCC-Reader-Komponente basiert auf dem PlainTextReader. Zu Kontrollzwecken (etwa Referenzierung oder Labeling) werden die ursprünglichen Satz-IDs in Tesla weitergegeben, so dass die Zuordnung auch in späteren Verarbeitungsschritten möglich ist. Neben dem Volltext stellt der Reader somit auch die einzelnen Sätze und deren IDs bereit. Da im SdeWaC-Korpus die wesentlichen Schritte der Vorverarbeitung (Tokenisierung, Stemming, POS-Tagging) bereits vollzogen wurden, kann die SdeWaC-Reader-Komponente direkten Zugriff auf die entsprechend qualifizierten Tokens geben, so dass neben den Sätzen auch die den einzelnen Token zugeordneten Annotationen für Wortstämme und Wortarten abgefragt werden können. Die in dieser Arbeit eingesetzten Korpora wurden in die verwendete Tesla-Version eingebunden, so dass die Experimente unmittelbar nachvollzogen werden können.[91]

6.2.2 Vorverarbeitung

Die Erstellung von Kookkurrenzvektoren setzt zunächst eine Vorverarbeitung der Korpusdaten voraus. Wie oben beschrieben werden in dieser Arbeit zwei verschiedene Korpusformate eingesetzt. Während das SdeWac-Korpus bereits auf verschiedenen Ebenen vorverarbeitet vorliegt, so dass es über die zugehörige Reader-Komponente Zugriff auf die Sätze und die einzelnen Token sowie auf die zugehörigen Wortstämme und Wortarten-Annotationen (Part-Of-Speech bzw. POS-Tags) erlaubt, müssen die entsprechenden Vorverarbeitungsschritte (bis auf die Satzgrenzenerkennung) für die Korpora der Leipzig Corpora Collection (LCC) erst noch durchgeführt werden. Hierfür stehen in Tesla folgende Komponenten zur Verfügung:

- SimpleTokenizer
- TreeTaggerWrapper
- SnowballStemmerWrapper

Der SimpleTokenizer ist ein einfacher Tokenizer auf Basis des Java BreakIterator.[92] Neben der Unterteilung der Sätze in Tokens unterscheidet der SimpleTokenizer auch zwischen Wörtern, Zahlen, und Satzzeichen. Der für das POS-Tagging im SdeWaC eingesetzte TreeTagger (Schmid 1994) wird in Tesla über die TreeTaggerWrapper-Komponente bereitgestellt, so dass auch die als reine Textdateien vorliegenden LCC-Korpora mit Wortarten ausgezeichnet werden können. Der TreeTaggerWrapper kann zudem als Stemmer für die Ermittlung der Wortstämme eingesetzt werden. Für das Stemming kann alternativ

91 Zur Einbindung weiterer Korpora siehe Schwiebert (2012, Kapitel 4.1.3).
92 Siehe https://docs.oracle.com/javase/7/docs/api/java/text/BreakIterator.html (Zugriff vom 21.02.2018).

auch die SnowballStemmerWrapper-Komponente eingesetzt werden. Der Snowball-Stemmer[93] ist die offizielle Weiterentwicklung des regelbasierten Porter-Stemmers (Porter 1980), der für eine Reihe verschiedener Sprachen zur Verfügung steht (unter anderem Deutsch, Englisch, Französisch, aber auch Italienisch oder Russisch).

6.2.3 Kookkurrenzvektoren

Bei einer Modellierung auf Grundlage des Word Space Model besteht der zentrale Verfahrensschritt in der Erstellung von Wortvektoren, die die Grundlage für die weitere Modellierung darstellen. In einem ersten Schritt werden zunächst mit der VectorGenerator-Komponente einfache Kookkurrenzvektoren erstellt. Zentrale Parameter sind die Fensterbreite und die Vektorlänge. Die Fensterbreite wird direkt in der Komponente festgelegt, wobei die Werte zwischen 1 (nur direkte Nachbarn) und maximal dem gesamten Satz liegen.[94] Bei breitem Fenster kann zusätzlich eine Nachbarschaftsgewichtung nach Vorbild des Hyperspace Analogue to Language (HAL) eingesetzt werden, bei der die näher liegenden Elemente höher gewichtet werden (vergleiche Lund/Burgess 1996). Die Vektorlänge ist dagegen von der Merkmalsauswahl abhängig. Diese ist in Tesla durch (optionale) Filterkomponenten realisiert, welche der Vektorerstellung vorgeschaltet sind. Die Filter legen anhand verschiedener Kriterien fest, welche Types akzeptiert oder ausgeschlossen werden sollen. Es stehen im Wesentlichen folgende Filtertypen zur Verfügung:

- FrequencyFilter
- POSFilter
- WordlistFilter

Filterkriterien sind somit unter anderem die Frequenz oder die Wortart, zudem ist es möglich, die zu filternden Elemente explizit über eine Wortliste anzugeben. Die Filter können unabhängig voneinander als Context Filter oder als Creation Filter eingesetzt werden: Als Creation Filter legen sie fest, für welche Types Vektoren erstellt werden, als Context Filter dienen sie der Beschränkung des Merkmalssets, indem sie festlegen, für welche Kontextelemente Kookkurrenz gezählt wird.[95]

93 Siehe http://snowball.tartarus.org (Zugriff vom 21.02.2018).

94 Größere, d.h. Satzgrenzen überschreitende Fenster sind aufgrund der Beschaffenheit der verwendeten Korpora nicht sinnvoll möglich, da diese aus Gründen des Copyrights in einzelne, nicht fortlaufende Sätze aufgeteilt vorliegen (vgl. dazu Abschnitt 6.2.1).

95 So ist es beispielsweise möglich, über den WordlistFilter bestimmte Wörter als Kontextelemente auszuschließen (z.B. Stoppwörter), oder aber explizit vorzugeben, für welche Wörter Vektoren erstellt werden sollen. Eine weitere Möglichkeit zur Einschränkung des Merkmalssets ist die Merkmalsauswahl mittels Wortart. Bei einem mit POS-Tags versehenen Korpus kann über den POSFilter bspw. festgelegt werden, dass nur Nomen zugelassen sind. Auf diese Weise kann die Kookkurrenz bspw. auf

Es können beliebig viele Filter eingesetzt und dabei frei kombiniert werden. Bei einem Verzicht auf jegliche Filterung werden Vektoren für alle Types erstellt, die vom Tokenizer (oder direkt vom Reader) geliefert werden. Dabei wird die Kookkurrenz gegenüber *allen* anderen Elementen gezählt, so dass die Vektorlänge der Gesamtanzahl der Types entspricht. Anstelle einer Filterung kann auch ein vollständiges Merkmalsset angegeben werden, was weitere Formen der vorherigen Merkmalsauswahl eröffnet, etwa den Einsatz einer vorab durch Wortlisten oder Ähnliches festgelegten Menge an Attributen.[96]

6.2.4 Normalisierung

Aufgrund der unterschiedlichen Auftrittshäufigkeiten der Wörter sind auch die Unterschiede hinsichtlich der Belegung der Kookkurrenzwerte zum Teil sehr groß. Dadurch weisen die rohen Kookkurrenzvektoren unterschiedliche Längen in Bezug auf den zugrunde gelegten Vektorraum auf. Um dies auszugleichen, ist es üblich, die Vektoren zu normalisieren, indem jedes Vektorelement durch die euklidische Länge des Vektors dividiert wird, berechnet als die Wurzel aus der Summe aller Vektorelemente:[97]

$$\left|\vec{v}_j\right| = \sqrt{\sum_{i=1}^{n} v_{i,j}^2}$$

Die Normalisierung ist zum einen als eigene Komponente realisiert. In der VectorNormalisation-Komponente kann zwischen einer Normalisierung nach euklidischer Länge und einer einfachen Variante nach Levy/Bullinaria (2001) gewählt werden, bei der die Normalisierung unter Berücksichtigung der Fensterbreite und der Frequenz erfolgt. Da die Normalisierung ein Standardschritt ist, der nur in bestimmten Fällen nicht eingesetzt werden kann (zum Beispiel wenn für die nachträgliche Gewichtung die ursprünglichen Kookkurrenzwerte benötigt werden), wurde die Funktionalität zum anderen auch mit in die Komponente zur Vektorgewichtung integriert,[98] die im folgenden Abschnitt beschrieben wird.

Subjekte und Objekte beschränkt werden, so dass der Wortraum zu einem gewissen Grade zu einem grammatisch ausgezeichneten Raum wird. Sowohl die POS-Filterung als auch der WordlistFilter wurden im Zuge dieser Arbeit als eigene Tesla-Komponenten implementiert, kommen in den hier beschriebenen Experimenten jedoch nicht zum Einsatz (dafür zum Beispiel in Richter u.a. 2015).

96 Da das Merkmalsset als Mapping hinterlegt ist, besteht zudem die Möglichkeit, mehrere Wörter auf ein gemeinsames Merkmal abzubilden, welches dann als eine Art (Äquivalenz-)Klasse verstanden wird. Eine mögliche Anwendung hierfür ist die Reduzierung des Merkmalssets (im Sinne einer Dimensionsreduktion) durch eine Clusteranalyse.

97 Formel wiedergegeben nach Manning u.a. (2008).

98 Die Integration ist demnach als eine verarbeitungstechnisch motivierte ›Abkürzung‹ zu verstehen, konzeptuell ist die Normalisierung als eigener Schritt anzusehen.

6.2.5 Gewichtung

Die Gewichtung von Vektorelementen hat das Ziel, das Verhältnis zwischen dem Kookkurrenten und dem beschriebenem Wort mit einzubeziehen. Mit der dem HAL-Modell entlehnten Nachbarschaftsgewichtung ist eine sehr einfache Form direkt in der VectorGenerator-Komponente integriert (siehe Abschnitt 6.2.3). Weitere Möglichkeiten stehen in der VectorWeighting-Komponente zur Auswahl, mit der eine nachträgliche Gewichtung der rohen Kookkurenzvektoren vorgenommen werden kann:[99]

- log-smoothing
- precedence
- tf.idf-Gewichtung
- Pointwise Mutual Information (PMI)
- Log-Likelihood-Ratio (LLR)

Die einfachste Form der Gewichtung ist das sogenannte »log-smoothing«, bei dem zu jedem Kookkurrenzwert der Logarithmus errechnet wird. Durch die Übertragung auf die logarithmische Skala wird der Wertebereich verengt, so dass starke Abweichungen gewissermaßen ›geglättet‹ werden, damit ein dreimaliges gemeinsames Auftreten nicht als dreifache Relevanz gegenüber der einmaligen Kookkurrenz gewertet wird. Die »precedence« entspricht dem von Sinclair (1991) beschriebenen Vorgehen zur Ermittlung von Kollokationen, bei dem der Kookkurrenzwert in Relation zur Gesamtfrequenz des Attributs gesetzt wird (siehe Sinclair 1991, 106). Die hier implementierte Variante einer tf.idf-Gewichtung kombiniert die beiden erstgenannten Gewichtungen.

Während die dem Information Retrieval entlehnte tf.idf-Gewichtung im Bereich der Wortvektoren eher unüblich ist und hier nur im Hinblick auf die Bereitstellung verschiedener Konfigurationsmöglichkeiten einbezogen wurde, handelt es sich bei der PMI und der LLR um informationstheoretisch motivierte Assoziationsmaße, mit denen die Signifikanz von Kookkurrenzen bewertet werden kann.[100] Für die Gewichtung von Wortvektoren ist die PMI eines der am weitesten verbreiteten Maße, hier wiedergegeben in der Formulierung von Church/Hanks (1990):

$$pmi(x, y) = \log_2 \frac{P(x, y)}{P(x)P(y)}$$

99 Weitere Formen der Gewichtung lassen sich aufgrund des in Tesla konsequent verfolgten API-Prinzips sehr einfach integrieren (vgl. Abschnitt 6.1).

100 Für eine ausführliche Herleitung der beiden Maße siehe Anhang C.

Durch die PMI wird die bedingte Wahrscheinlichkeit des gemeinsamen Auftretens zweier Wörter x und y in Relation zu ihren jeweiligen Auftrittswahrscheinlichkeiten gesetzt. Die LLR nach Dunning (1993), auch als G^2-Test bekannt, ermittelt dagegen den Grad der Wahrscheinlichkeit (*likelihood*), ob es sich bei dem gemeinsamen Vorkommen um ein abhängiges oder ein unabhängiges Ereignis handelt.

Im Kontext dieser Arbeit übernehmen die Assoziationsmaße eine doppelte Funktion: In der VectorWeighting-Komponente dienen sie der Gewichtung der Kookkurrenzvektoren, um damit signifikante Kookkurrenzen stärker zu betonen; gleichzeitig können die hier verwendeten Assoziationsmaße auch für die Bewertung lokaler Kontexte verwendet werden, wie sie im Rahmen dieser Arbeit im Zusammenhang mit der Erstellung von Kontextvektoren durchgeführt wird – dies wird im Folgenden beschrieben.

6.2.6. Token-Vektoren

Die Erstellung von Vektoren auf Grundlage einzelner Verwendungen bildet den Kern der im Rahmen dieser Arbeit vorgenommenen Modellierung der Bedeutungskonstitution. Grundlage bildet das in Kapitel 5 zugrunde gelegte Prozessschema, mit dem die Bedeutungskonstitution in Anlehnung an Marr (1982) als informationsverarbeitender Prozess beschrieben wird. Gemäß diesem Schema wird eine Eingabeinformation in eine durch den Prozess transformierte Ausgabeinformation überführt. Wie in Abschnitt 5.1 beschrieben, besteht die Eingabeinformation für den Prozess dabei aus dem Kookkurrenzvektor eines Zielworts sowie den Vektoren der in dessen Kontexten jeweils auftretenden Elemente. Im Zuge des Prozesses wird der Vektor des Zielworts durch die Vektoren der Kontextelemente in einen neuen Vektor transformiert. Diese Ausgabeinformation repräsentiert dabei immer genau ein Vorkommen des Zielworts (das heißt genau ein Token) in einem lokalen Kontext. In dieser Arbeit wurden zwei Varianten solcher Token-Vektoren als austauschbare, separate Komponenten umgesetzt:

- ContextVectors
- CollocationVectors

In der konkreten Umsetzung beider Komponenten wird über die verschiedenen Kontexte eines Zielwortes iteriert: Für jeden Kontext werden die benötigten Kookkurrenzvektoren, die in den vorherigen Verarbeitungsschritten erstellt wurden, zunächst gesammelt und anschließend schrittweise in einen einzelnen Token-Vektor überführt. Das zu analysierende Zielwort wird über die Konfiguration festgelegt; optional können auch mehrere Zielwörter angegeben werden, deren Kontexte dann separat durchlaufen werden. Über einen entsprechenden Parameter kann zudem angegeben werden, ob jeweils alle Vorkommen betrachtet

werden oder ob nur eine begrenzte Anzahl an Kontexten verarbeitet werden soll. In einem weiteren Parameter kann die Kontextbreite festgelegt werden, das heißt die Anzahl der Kontextelemente, die berücksichtigt werden.[101]

Die beiden Komponenten unterscheiden sich im Wesentlichen darin, wie der Prozess der Transformation realisiert ist: In der ContextVectors-Komponente orientiert sich die Umsetzung des Prozesses weitgehend an dem von Schütze (1998) beschriebenen Vorgehen (siehe dazu auch Abschnitt 5.2). Anders als dort wird hier jedoch nicht der Zentroid über alle Vektoren im Kontext errechnet, sondern zunächst nur für die Kontextelemente, um ihn erst anschließend mit dem Zielwort zusammenzuführen, wodurch der Vektor des Zielworts ein höheres Gewicht gegenüber den Kontextelementen behält.[102] In der CollocationVectors-Komponente wird der Prozess zusätzlich um eine vorherige Gewichtung der Kontextelemente ergänzt. Für die Bewertung der Kontextelemente kann zwischen der Pointwise Mutual Information (PMI) und der Log-Likelihood-Ratio (LLR) gewählt werden, die auch in der Gewichtung der Vektoren eingesetzt werden (siehe Abschnitt 6.2.5). Als Folge der Gewichtung wird der Zentroid hier nur über die signifikantesten Kontextelemente berechnet und erst in einem weiteren Schritt mit dem Kookkurrenzvektor des Zielworts zusammengeführt. Dies entspricht dem in Abschnitt 5.2.2 beschriebenen erweiterten Prozess, so dass die CollocationVector-Komponente als direkte softwaretechnologische Realisierung der in dieser Arbeit vorgenommenen Modellierung der Bedeutungskonstitution anzusehen ist. Die resultierenden Token-Vektoren können zum einen an weitere Verarbeitungsschritte übergeben werden, zum anderen besteht in den Komponenten die Möglichkeit, die Token-Vektoren direkt zu visualisieren (siehe dazu Abschnitt 6.2.8); hierbei kann festgelegt werden, wie viele der Vektoren geplottet werden sollen.

6.2.7 Clusteranalyse

Wie die Gewichtung übernimmt auch die Clusteranalyse bei der Modellierung zwei unterschiedliche Funktionen: zum einen ist es in einigen Experimenten nötig, Gruppen von Wortvektoren bzw. Kontextvektoren zu erstellen. Hierbei wird ein sogenanntes flaches Clustering eingesetzt, das die analysierten Elemente in verschiedene Cluster einteilt, ohne diese untereinander in Beziehung zu setzen. Zum anderen wird für die Ergebnisbewertung eine hierarchische Clusteranalyse eingesetzt, bei der auch die Beziehungen zwischen den gefundenen Gruppierungen berücksichtigt werden. Da die hierarchischen Verfahren vor allem für die

101 Da die verwendeten Korpora keine fortlaufenden Texte enthalten, ist die Kontextbreite maximal auf die jeweilige Satzlänge beschränkt.

102 Motiviert ist dieses Vorgehen durch die zugrunde gelegte theoretische Konzeption, der zufolge eine lokale Aktivierung von Teilen des Bedeutungspotentials erfolgt, welches durch den Kookkurrenzvektor des Zielworts repräsentiert wird (vgl. dazu Abschnitt 5.2.1).

Visualisierung eine Rolle spielen, werden sie an entsprechender Stelle dieses Kapitels beschrieben (siehe Abschnitt 6.2.8).

Für die Anwendung verschiedener Formen der flachen Clusteranalyse wurde die ELKI-API[103] eingebunden (Achtert u.a. 2012). ELKI setzt Indexstrukturen ein, was eine um ein Vielfaches schnellere Verarbeitung im Vergleich zu anderen Data-Mining-Frameworks wie zum Beispiel WEKA oder auch R ermöglicht.[104] Im Rahmen dieser Arbeit wurden über die ELKI-API insgesamt acht Algorithmen in Tesla bereitgestellt:

- K-Means-MacQueen
- K-Means-Lloyd
- K-Medians-Lloyd
- K-Medoids-EM
- K-Medoids-PAM
- DBSCAN
- OPTICS
- SNN

Bei den fünf erstgenannten Algorithmen handelt es sich um sogenannte distanzbasierte Verfahren: Neben der klassischen Implementation des K-Means-Algorithmus nach Lloyd (1982) sowie der gleichnamigen Variante nach Macqueen (1967) sind dies im Wesentlichen Erweiterungen, die sich vor allem in der Berechnung der Clusterzentren unterscheiden. Zusätzlich wurden mit DBSCAN[105] (Ester u.a. 1996) und dessen Weiterentwicklung OPTICS[106] (Ankerst u.a. 1999) auch zwei dichtebasierte Verfahren integriert, sowie mit dem Shared-Nearest-Neighbor-Clustering (SNN) ein auf dem DBSCAN-Algorithmus aufbauendes Verfahren, das die Konzepte von Dichte und Distanz kombiniert (siehe dazu Ertöz u.a. 2003). Die meisten der genannten Algorithmen werden in dieser Arbeit nicht genutzt; wie schon in den vorangegangenen Verfahrensschritten wurde jedoch auch hier der Gedanke verfolgt, eine Austauschbarkeit der Verfahren zu gewährleisten, etwa um sie für weiterführende Analysen einzusetzen. Weil die distanzbasierten Verfahren bei hochdimensionalen Daten als problematisch gelten,[107] wird in den Experimenten vorzugsweise der DBSCAN-Algorithmus eingesetzt. Ein (gewünschter) Seiteneffekt ist dabei, dass

103 Siehe https://elki-project.github.io (Zugriff vom 21.02.2018). Die hier verwendete Version 0.5.5 ist auf den 10.12.2012 signiert.

104 Für entsprechende Vergleichstests siehe https://elki-project.github.io/benchmarking (Zugriff vom 21.02.2018).

105 Density-Based Clustering of Applications with Noise.

106 Ordering Points To Identify the Clustering Structure.

107 In diesem Zusammenhang wurde von Bellmann (1961) der Begriff des »curse of dimensionality« geprägt, da sich das Volumen bei steigender Dimensionalität exponentiell vergrößert. Das Problem betrifft dabei u.a. auch die Definition dessen, was ein nächster Nachbar (»nearest neighbor«) ist, da die Abstände zwischen den Elementen in höher dimensionierten Räumen unter Umständen extrem

die Clusterzahl (k) in DBSCAN nicht fest vorgegeben werden muss, sondern erst im Zuge der Verarbeitung ermittelt wird.

6.2.8. Visualisierung

Die (Zwischen-)Ergebnisse der einzelnen Verfahrensschritte bestehen im Kontext dieser Arbeit meist aus Mengen von Vektoren, die zum Teil zusätzlich mittels Clusteranalyse gruppiert werden. Um die Interpretation der Ergebnisse zu erleichtern, wurden verschiedene Möglichkeiten zur Visualisierung in Tesla integriert. Grundlage der Visualisierung ist die frei verfügbare Statistik-Software R.[108] R stellt eine Vielzahl von Funktionen für die statistische Analyse und die Visualisierung komplexer Datensätze bereit; gleichzeitig ist R auch eine eigene Programmiersprache, die eine sehr kompakte Formulierung der für die Visualisierung nötigen Datenkonversionen und statistischen Operationen erlaubt. Im Zuge dieser Arbeit wurden drei Visualisierungs-Typen in Tesla integriert:

- Scatterplots
- Dendrogramme
- Phylogenetische Bäume

Die Erstellung von Scatterplots ist eine der Basisfunktionen von R. In Scatterplots werden die Elemente eines Datensatzes auf ein zwei- bzw. dreidimensionales Raster abgebildet (siehe Abb. 6.5, oben). Die hierfür nötige Dimensionsreduktion wird direkt in R mittels Multidimensionaler Skalierung (MDS) durchgeführt, ein Verfahren der multivariaten Statistik, bei dem die Objekte möglichst topologieerhaltend in einen Datenraum geringerer Dimensionalität überführt werden. Da es bei der Auswertung der Ergebnisse oftmals hilfreich ist, den Datenraum in strukturierter Form darzustellen, können zudem verschiedene Formen von Dendrogrammen (siehe Abb. 6.5, unten) geplottet werden. Dendrogramme sind Baumdarstellungen, die auf einem vorherigen hierarchischen Clustering der Daten basieren. Die hierfür in R bereitgestellte Funktion setzt standardmäßig die UPGMA-Methode[109] (Sokal/Michener 1958) ein.

Zwischen Scatterplots und Dendrogrammen lässt sich zudem eine direkte Beziehung herstellen: Je nachdem in welcher Höhe man das Dendrogramm ›schneidet‹ (horizontale Linie in Abb. 6.5 unten), erhält man jeweils ein flaches Clustering, das parallel zum Dendrogramm in einem Scatterplot dargestellt werden kann. Die entsprechenden Clusterzugehörigkeiten können dabei farblich

voneinander abweichen können, wodurch das Konzept der Nähe unterlaufen wird (siehe dazu auch Sahlgren 2006, 20f.).

108 Siehe https://www.r-project.org (Zugriff vom 21.02.2018).

109 Unweighted Pair Group Method with Arithmetic mean.

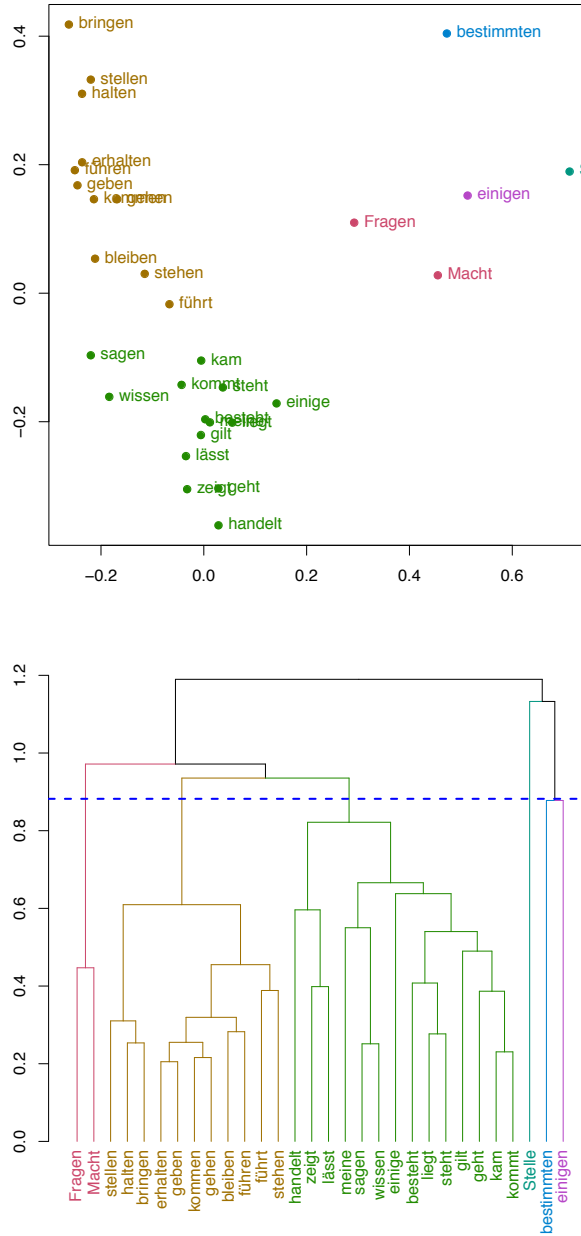


Abbildung 6.5: Beispiele für die Visualisierung eines Ausschnitts des Wortraums. Scatterplots projizieren die Datensätze auf ein zweidimensionales Raster; in Dendrogrammen werden die Daten durch ein hierarchisches Clustering in eine Baumstruktur überführt.

hervorgehoben werden. Neben Dendrogrammen können die Daten auch in Form von phylogenetischen Bäumen visualisiert werden, die verschiedene zusätzliche Layout-Varianten erlauben (unter anderem auch kreisförmige Darstellungen sowie sogenannte »unrooted trees«, also Bäume ohne Wurzelknoten). Hierfür wurde das R- Programmpaket ape[110] eingesetzt, das verschiedene Standardmethoden der Bioinformatik bereitstellt. Über das ape-Paket wurde mit dem Neighbor-Joining-Algorithmus (Saitou/Nei 1987) zudem eine zusätzliche Alternative zum UPGMA-Clustering integriert.

Die Plotting-Funktion steht zum einen direkt in den Komponenten zur Vektorerzeugung und -manipulation zur Verfügung, zum anderen wurde sie auch in der RPlotter-Komponente gekapselt, die eine Reihe von zusätzlichen Parametern bietet: hier besteht unter anderem die Möglichkeit, einzelne Wörter gezielt im Plot hervorzuheben; weitere Optionen sind zum Beispiel die Markierung der im hierarchischen Clustering gefundenen Gruppen durch Farben oder Boxen. Auch in den Cluster-Komponenten besteht die Möglichkeit, Plots direkt zu generieren. Zum einen können die einzelnen Cluster geplottet werden, etwa um deren interne Struktur zu verdeutlichen. Zum anderen besteht die Möglichkeit, das vollständige Clusterergebnis in einem einzelnen Scatterplot zu visualisieren, in dem die durch die Clusteranalyse gefundenen Clusterzuordnungen farblich hervorgehoben sind. Bei allen Varianten wird das verwendete Skript zusammen mit dem Plot gespeichert, so dass auch eine nachträgliche Anpassung des Plots möglich ist.

6.2.9 Beispielwörter für die Experimente

Ziel der Experimente ist eine Simulation der Bedeutungskonstitution. Vor dem Hintergrund der Annahme, dass sich der Prozess bei heterogenen Kontexten besonders deutlich nachweisen lässt, sollen für die Durchführung der Experimente vor allem mehrdeutige Beispielwörter eingesetzt werden. Als Quelle für die Auswahl der Untersuchungsbeispiele wird in dieser Arbeit die GermaNet-Datenbank eingesetzt.[111] GermaNet ist ein an der Universität Tübingen entwickeltes lexikalisch-semantisches Wortnetz für das Deutsche, das nach dem Vorbild von WordNet[112] strukturiert ist (siehe Hamp/Feldweg 1997; Henrich/Hinrichs 2010). GermaNet enthält Einträge für Nomen (N), Verben (V) und Adjektive (A), die über semantische Relationen wie Hyponymie oder Antonymie verknüpft

110 Analyses of Phylogenetics and Evolution, siehe https://cran.r-project.org/web/packages/ape/index.html (Zugriff vom 21.02.2018).

111 Siehe http://www.sfs.uni-tuebingen.de/GermaNet (Zugriff vom 21.02.2018). Die Entscheidung für GermaNet ist dabei als mehr oder weniger arbiträr anzusehen – eine Alternative wäre, in einem gängigen Wörterbuch nach der Anzahl der Haupteinträge zu gehen, da diese in der Regel für verschiedene Lesarten und somit für mögliche (Teil-)Bedeutungen von Wörtern stehen.

112 Siehe http://wordnet.princeton.edu (Zugriff vom 21.02.2018).

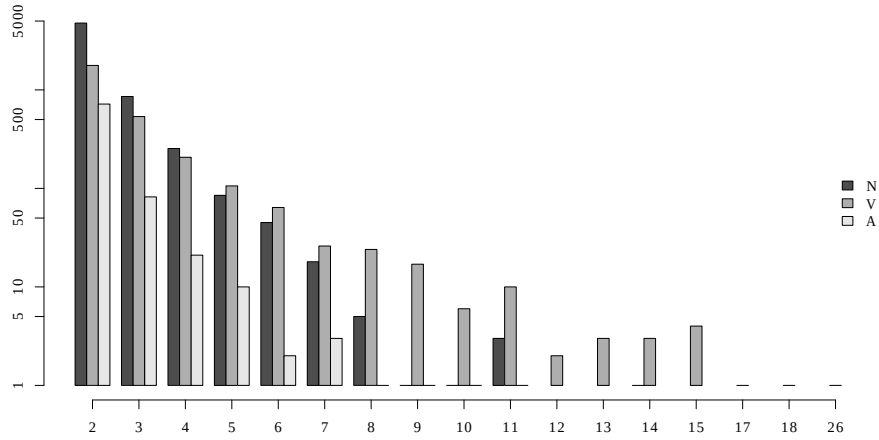


Abbildung 6.6: Verteilung mehrdeutiger Einträge in GermaNet, unterteilt nach Wortart. Mehrdeutigkeit äußert sich in GermaNet in der Anzahl der Synsets, denen ein Eintrag zugeordnet ist; die Höhe der Balken zeigt die jeweilige Anzahl der Einträge.

sind. Die Einträge sind in sogenannten Synsets organisiert, die neben möglichen Schreibweisen auch eine Liste synonym verwendeter Wörter enthalten. Ist ein Eintrag mehreren solcher Synsets zugeordnet, so kann daraus auf einen mehrdeutigen Gehalt geschlossen werden. Dies ist bei ca. 10% der in GermaNet repräsentierten Wörter der Fall,[113] deren Verteilung hinsichtlich der Anzahl von Synsets in Abb. 6.6 wiedergegeben ist.

Im Hinblick auf die Beispielwörter ergibt sich daraus als Auswahlkriterium, dass diese möglichst vielen Synsets zugehören sollten, da dies auf ein heterogenes Verwendungsprofil schließen lässt.

Abb. 6.7 zeigt jeweils die zehn Wörter mit den meisten Synset-Zuordnungen für die drei in GermaNet erfassten Hauptwortarten. Unter den Adjektiven finden sich dabei einige Beispiele, die auch als Verb verwendet werden können (*übertragen, ergeben, verfallen, versehen, wollen*). Abb. 6.8 fasst die in GermaNet enthaltenen Fälle einer solchen lexiko-syntaktischen Ambiguität zusammen.

Um eine geeignete Auswahl für die Experimente zu erhalten, ist ein Abgleich mit den verwendeten Korpora nötig. Neben einer möglichst hohen Zahl von Synset-Zuordnungen sollten auch möglichst viele Belegstellen vorliegen, um eine solide statistische Grundlage für die Analyse zu erhalten. Auf dieser Grundlage wurden für jede Wortart (einschließlich der Menge der V/A-ambigen Wörter) möglichst repräsentative Beispiele ausgewählt, zusammengefasst in Tabelle 6.1.

113 In der hier genutzten Version 9.0 (April 2014) enthält GermaNet 93.246 Synsets; von den 121.810 in GermaNet repräsentierten Wörtern sind 9.625 mindestens 2 Synsets zugeordnet.

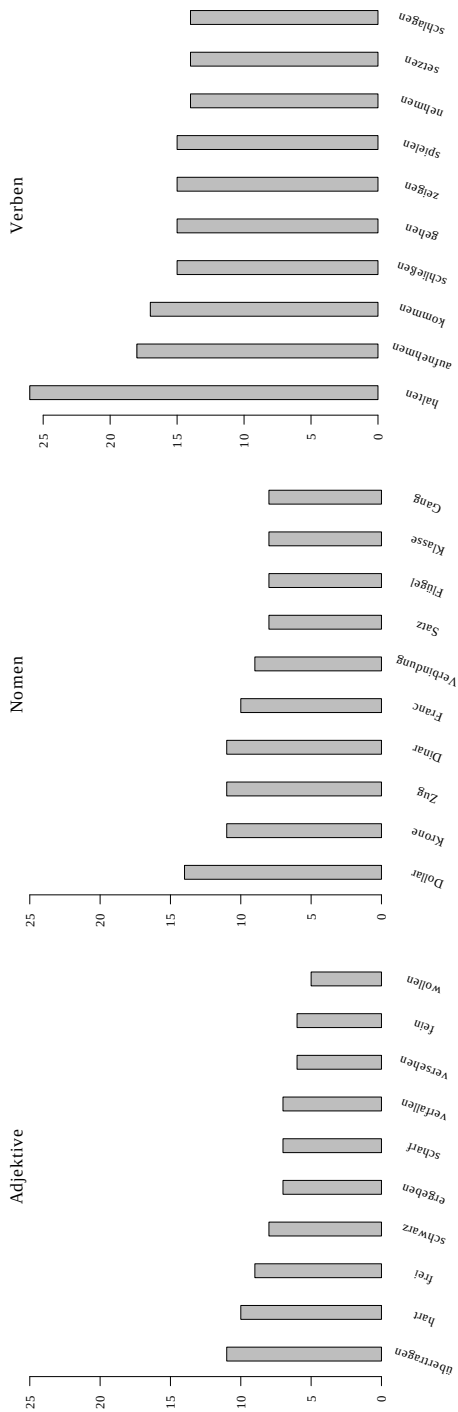


Abbildung 6.7: Mehrdeutige Einträge in GermaNet, sortiert nach der Anzahl der Synset-Zuordnungen (unterteilt nach Wortart).

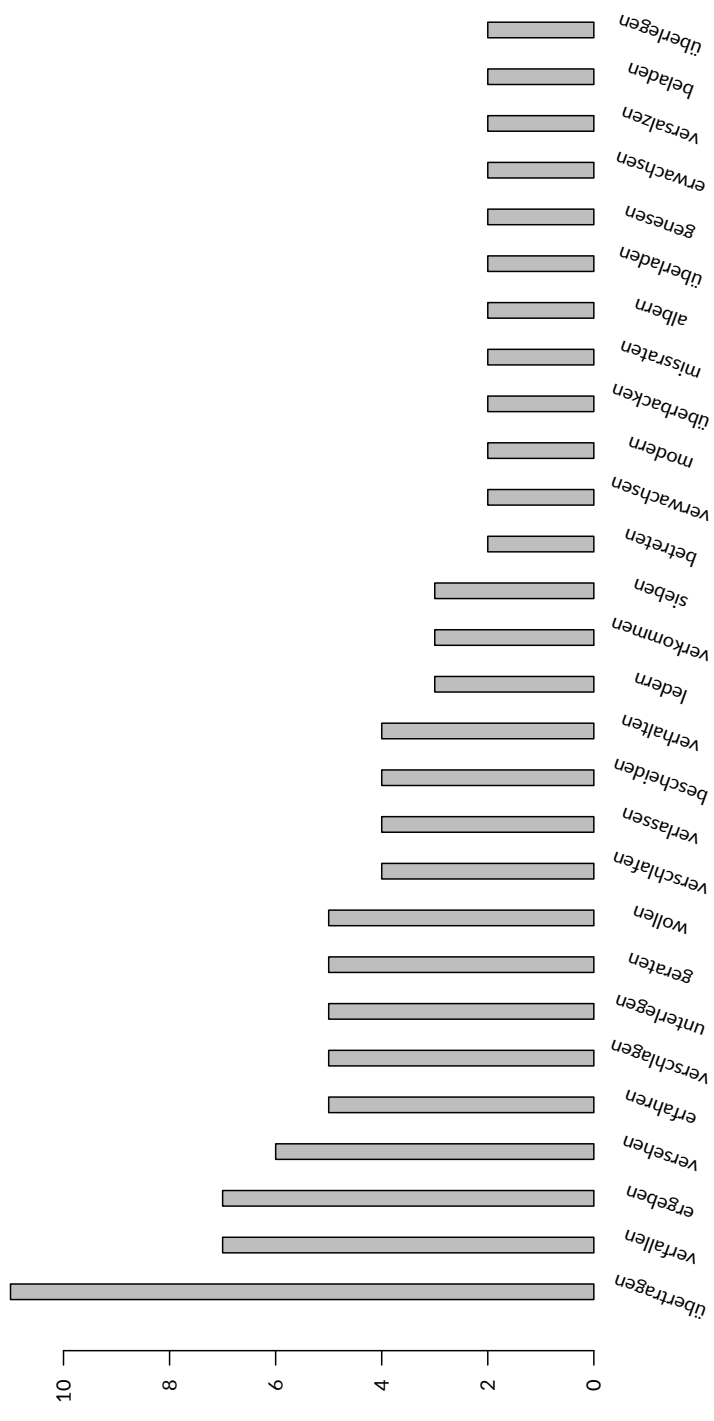


Abbildung 6.8: Übersicht der Einträge, die in GermaNet sowohl mehreren Synsets als auch mehreren Wortarten zugeordnet sind (A und V).

	Wort	LCC	SdeWaC	Synsets
V	halten	3329	6171	26
	spielen	2575	3494	15
N	Klasse	968	2340	8
	Krone	212	377	11
A	hart	685	789	10
	scharf	402	470	7
A/V	verhalten	354 (88/266)	789 (83/706)	4
	erwachsen	113 (19/94)	309 (29/280)	2

Tabelle 6.1: Häufigkeiten der ausgewählten Beispiele in den beiden Korpora, im Falle der kategorial ambigen Wörter unterschieden nach Wortart (A/V).

Im Sinne der Auswahlkriterien sind vor allem die Verben als gute Beispielwörter anzusehen. Es fällt auf, dass hier offenbar deutlich häufiger eine Mehrfachzuordnung vorgenommen wird,[114] was sich mit der in Abschnitt 2.1 angeführten Beobachtung deckt, dass Verben besonders häufig über einen mehrdeutigen Gehalt verfügen. Über diesen lässt sich auch die relativ hohe Frequenz in den Korpora deuten, die sich aus den entsprechend flexibleren Verwendungsmöglichkeiten ergibt.

6.3 Zusammenfassung

In diesem Kapitel wurden zunächst die zentralen Konzepte des linguistischen Komponentensystems Tesla vorgestellt. Darauf aufbauend wurden anschließend die im Zuge dieser Arbeit umgesetzten Erweiterungen in Bezug auf die Komponentenausstattung beschrieben, die für eine Modellierung der Bedeutungskonstitution mithilfe des Word Space Model (WSM) nötig sind. Tesla dient in dieser Arbeit als virtuelles Labor, über das eine Arbeitsumgebung für die Erstellung und Durchführung von sprachwissenschaftlichen Experimenten bereitgestellt wird.

Für die Komponentenentwicklung steht mit der Developer Perspective eine vollwertige Java-IDE zur Verfügung. Bei der Konzeption und Umsetzung von Komponenten kann dabei auf den vollständigen Funktionsumfang der zugrunde gelegten Programmiersprache Java zurückgegriffen werden: Zum einen können

114 Tatsächlich ist fast ein Drittel der Verben mehreren Synsets zugeordnet (2.782 von 9.340); bei Nomen ist das Verhältnis mit knapp 7% dagegen deutlich geringer (6.038 von 85.662), ebenso bei Adjektiven mit 6,5% (840 von 12.890).

Komponenten durch die Aggregation von Methoden und Datentypen in Form von Klassen und Interfaces realisiert werden, zum anderen können die in der Verarbeitung erzeugten Datenstrukturen direkt als (beliebig komplexe) Java-Objekte weitergegeben werden. Ein wesentlicher Vorteil besteht darin, dass mittels des Tesla Role System (TRS) auch die auf die jeweiligen Datenstrukturen zugeschnittenen Zugriffsmethoden mit weitergegeben werden können. Dies ist insbesondere bei komplexeren Datenstrukturen von Interesse, wie bei den im Kontext dieser Arbeit verwendeten Kookkurrenzvektoren oder auch bei den auf Grundlage der Vektoren erstellten Clustern. Für das experimentelle Arbeiten stellt Tesla mit der Linguist Perspective eine eigene Umgebung zur Verfügung, in der unter anderem die Komponenten in einem graphischen Editor zu Experimenten arrangiert werden können. Ein Tesla-Experiment umfasst dabei zum einen eine Spezifikation des vollständigen Versuchsaufbaus im virtuellen Labor, zum anderen auch das Protokoll seiner Ausführung, einschließlich aller Zwischenergebnisse.

Mit dieser Art der umfassenden Dokumentation entspricht der in Tesla umgesetzte Experimentbegriff den wesentlichen Anforderungen, die auch seitens der Kognitiven Linguistik an Experimente gestellt werden und unter anderem in der Möglichkeit zur Kontrolle, Wiederholbarkeit und Variation von Experimenten bestehen (siehe Rickheit u.a. 2010, 196). Im Vordergrund steht dabei vor allem die Schaffung von kontrollierten Bedingungen bei der Durchführung der Experimente, welche durch die vollständige Spezifizierung aller relevanten Parameter gegeben ist. Durch die umfassende Dokumentation wird gleichzeitig die Anforderung der Wiederholbarkeit von Experimenten adressiert. Die Tesla-Experimente werden als XML-Dokumente verwaltet, die exportiert und an anderer (oder gleicher) Stelle wieder importiert und erneut durchgeführt werden können, etwa um experimentelle Ausgänge zu überprüfen: solange die verwendeten Bestandteile (Datenquellen, Komponenten) verfügbar sind, ist sichergestellt, dass die Umgebungsbedingungen identisch zur ursprünglichen Umgebung sind. Die von Rickheit u.a. (2010, 196) als dritte Anforderung genannte Möglichkeit zur Variation der Experimente basiert wesentlich auf dem in Tesla umgesetzten Komponentenmodell, das eine Typisierung der Schnittstellen mittels des TRS anhand ihrer funktionalen Rolle vorsieht. Durch den modularen Aufbau der Tesla-Experimente in Form von Komponenten-Workflows können die Verfahren zum Beispiel auf eine andere Datenbasis angewendet, in ihrer Anordnung verändert oder durch eine Modifikation der Parameterkonfigurationen der eingesetzten Komponenten variiert werden, um dadurch die Methoden zu evaluieren und gegebenenfalls zu optimieren. Hierbei kommt zusätzlich ein weiterer Vorteil der vollständigen Speicherung der Zwischenergebnisse zum Tragen: Bei der erneuten Durchführung eines Experiments müssen die Komponenten nur dann neu ausgeführt werden, wenn sich ihre Konfiguration (oder die einer vorgeschalteten Komponente) verändert hat, da sich damit unter Umständen die weitergegebenen Daten verändern können. Dies begünstigt die in dieser Arbeit verfolgte empirisch-experimentelle Herangehensweise, da einzelne Komponenten

ausgetauscht oder ihre Parameter variiert werden können, ohne dass anschließend das gesamte Experiment erneut ausgeführt werden muss. Da dies zudem experimentübergreifend funktioniert, müssen bspw. Korpora nicht mehrfach eingelesen und vorverarbeitet werden, was vor allem bei größeren Datenmengen oder sehr rechenintensiven Verarbeitungsschritten die Verarbeitungsdauer deutlich verkürzt und damit eine erhebliche Erleichterung darstellt.

Aufgrund seiner offenen Konzeption und seines flexiblen Komponentenmodells ist Tesla in hohem Maße erweiterbar. Damit Tesla in dieser Arbeit eingesetzt werden kann, wurde eine Reihe zusätzlicher Komponenten umgesetzt (siehe Abschnitt 6.2). Speziell für den Umgang mit Vektoren und Clustern wurde zudem eine Plotting-Funktion implementiert, die bei Bedarf direkt über die Komponenten eingesetzt werden kann und die es ermöglicht, die erzeugten Daten zu visualisieren, so dass die Ergebnisse leichter interpretiert werden können. Die Erweiterungen dienen in erster Linie als Grundlage für die konkrete Umsetzung der im vergangenen Kapitel beschriebenen Modellierung der Bedeutungskonstitution, die wesentlich auf der Verwendung von Wortvektoren aufbaut. Mit der Bereitstellung von Komponenten und Workflows in Form von Experimenten kann Tesla jedoch auch über den Rahmen dieser Arbeit hinaus als virtuelles Labor für die Bearbeitung von Fragestellungen der Kognitiven Linguistik dienen. Da im Rahmen dieser Arbeit die distributionelle Methodik des WSM adaptiert wird, kann Tesla abseits des konkreten Anwendungsfalls auch ganz allgemein für distributionell motivierte Experimente eingesetzt werden.[115] Damit wird hier gleichzeitig ein weiteres Ziel dieser Arbeit eingelöst, das wie eingangs formuliert in der Schaffung einer Arbeitsumgebung zur Durchführung von Experimenten in einem distributionellen Framework besteht.

Im Mittelpunkt dieser Arbeit steht jedoch die Modellierung der Bedeutungskonstitution auf Grundlage des WSM. Die hierfür vorgesehenen konkreten Experimente werden im nachfolgenden Kapitel beschrieben. Zusätzlich zu den in diesem Kapitel genannten Komponenten, die als Grundbausteine für die Umsetzung der Modellierung angesehen werden können, werden dort zum Teil weitere, spezialisierte Komponenten eingesetzt, die auf den hier beschriebenen basieren. Diese werden bei Bedarf im Kontext der jeweiligen Experimente beschrieben.

115 So wurden beispielsweise einige der in diesem Kapitel beschriebenen Komponenten eingesetzt, um die Aspektklassen-Typologie nach Vendler (1967) experimentell zu rekonstruieren (vgl. dazu Richter u.a. 2015).

7. Experimente zur Bedeutungskonstitution

In diesem Kapitel wird die konkrete Umsetzung der in Kapitel 5 vorgenommenen Modellierung des Prozesses der Bedeutungskonstitution beschrieben. Die Umsetzung des Modells erfolgt in einer Reihe von Experimenten, die jeweils beispielhaft für verschiedene Wörter durchgeführt werden. Ziel ist es, das Modell anhand der Beispielanalysen experimentell zu überprüfen, um daraus Rückschlüsse auf die zugrunde gelegten theoretischen Annahmen ziehen zu können. Maßgeblich ist hierbei die in Abschnitt 5.3 formulierte Erwartung: wenn es möglich ist, die Bedeutungsvariation in den Experimenten durch eine kontextuell bedingte Veränderung der Repräsentation sichtbar zu machen und dies mit sinnvoll interpretierbaren Veränderungen der Ähnlichkeit zu anderen Repräsentationen einhergeht, dann kann dies vor dem Hintergrund der theoretischen Annahmen als Indikator für eine erfolgreiche Modellierung des Prozesses der Bedeutungskonstitution angesehen werden.

In Abschnitt 7.1 wird der experimentelle Aufbau zur Erstellung von Kookkurrenzvektoren in Tesla beschrieben, der als Basis für die weiteren Experimente in diesem Kapitel dient. Anschließend werden auf Grundlage der Kookkurrenzvektoren kleinere Ausschnitte des Wortraums erstellt, in denen zu einem Zielwort die ähnlichsten Elemente zusammengefasst sind. Diese können in der Folge als Referenzräume für die Visualisierung der Bedeutungskonstitution eingesetzt werden. In Abschnitt 7.2 wird in einem darauf aufbauenden Experiment gezeigt, wie durch Hinzunahme der Kontexte auch Einzelvorkommen kodiert werden können und wie sich dadurch die Repräsentation verändert. Hier erfolgt die eigentliche Umsetzung des Prozesses der Bedeutungskonstitution. Der Kookkurrenzvektor eines Zielworts wird hierbei mit den Vektoren der Kontextelemente kombiniert, um einen neuen Vektor zu bilden, der die jeweilige Kontextualisierung repräsentiert. Dieser kann im Anschluss gemeinsam mit dem Ausgangsvektor in einen Referenzraum projiziert werden, um die Veränderung sichtbar zu machen. Im abschließenden Experiment werden die lokal erzeugten Vektoren zueinander in Beziehung gesetzt (Abschnitt 7.3) und mittels Clusteranalyse strukturiert. Aus dieser Ausdifferenzierung des Bedeutungspotentials ergibt sich für jedes Wort ein semantisches Profil, das dessen Bedeutungsmöglichkeiten widerspiegelt. In Abschnitt 7.4 wird das Vorgehen in den Experimenten nochmals zusammengefasst und die Ergebnisse vor dem Hintergrund der theoretischen Vorannahmen diskutiert.

7.1. Repräsentation der Eingabeinformation

Ausgangspunkt für die Umsetzung des in Kapitel 5 beschriebenen Modells der Bedeutungskonstitution ist die Erstellung von Kookkurrenzvektoren, die gemäß

der Argumentation in Abschnitt 5.1 in der hier eingenommenen Perspektive zunächst nur das unausgedeutete Bedeutungspotential eines Wortes repräsentieren. Diese dienen in den nachfolgenden Experimenten einerseits als Eingabeinformation für den Prozess der Bedeutungskonstitution, andererseits können sie als Referenzgröße für die durch den Prozess hervorgerufene Veränderung der Repräsentation genutzt werden. In Abschnitt 7.1.1 wird zunächst der Aufbau des Experiments zur Erstellung von Kookkurrenzvektoren beschrieben, der gleichzeitig Bestandteil aller darauf aufbauenden Experimente ist. Anschließend wird die im Rahmen dieser Arbeit gewählte Parametrisierung erläutert (Abschnitt 7.1.2). Auf Grundlage der Kookkurrenzvektoren kann schließlich jeweils ein Referenzraum für die zu untersuchenden Wörter erstellt werden (Abschnitt 7.1.3), der als Vergleichsgröße in den nachfolgenden Experimenten zur Bedeutungskonstitution dient. Zudem wird hier auch anhand von Beispielkonfigurationen die Auswirkung der Parameter auf die Art der im Wortraum erfassten Ähnlichkeit illustriert.

7.1.1 Aufbau des Experiments

Abb. 7.1 zeigt den schematischen Workflow zur Erstellung von Kookkurrenzvektoren in Tesla. Zunächst wird ein Korpus, bestehend aus den ersten 1 Million Sätzen des SdeWaC-Korpus, durch eine entsprechende Reader-Komponente eingelesen (siehe Abschnitt 6.2.1). Da das SdeWaC-Korpus bereits vorverarbeitet vorliegt, kann hier auf die entsprechenden Schritte der Vorverarbeitung (Tokenisierung, Stemming, POS-Tagging) verzichtet werden.[116] Anschließend werden in der TF/IDF-Komponente die Token-Häufigkeiten ermittelt, so dass sie als Filterkriterium für die Merkmalsauswahl genutzt werden können. So kann einerseits über einen ContextFilter festgelegt werden, dass nur Wörter ab einer bestimmten Frequenz als Merkmale zugelassen werden, andererseits kann über einen CreationFilter gesteuert werden, für welche Wörter tatsächlich Vektoren erstellt werden.

Bei der Vektorerstellung werden sämtliche Sätze des Korpus linear durchlaufen, um zu den zuvor identifizierten Types die jeweiligen Kookkurenzen zu sammeln. Abschließend werden die Vektoren in der VectorWeighting-Komponente anhand ihrer euklidischen Länge normalisiert. Neben der Längennormalisierung bietet die Komponente verschiedene Möglichkeiten zur Gewichtung der Vektoren (siehe dazu Abschnitt 6.2.5).

116 Das ebenfalls 1 Million Sätze umfassende Korpus aus der Leipzig Corpus Collection (LCC), das in den Experimenten zu Vergleichszwecken eingesetzt wird, muss dagegen zunächst mit der SimpleTokenizer-Komponente in einzelne Token zerlegt werden; zusätzlich wird mit der SnowballStemmerWrapper-Komponente ein Stemming vorgenommen. Optional kann mit der TreeTaggerWrapper-Komponente hier auch ein POS-Tagging durchgeführt werden. Eine Beschreibung der entsprechenden Komponenten findet sich in Anhang A.

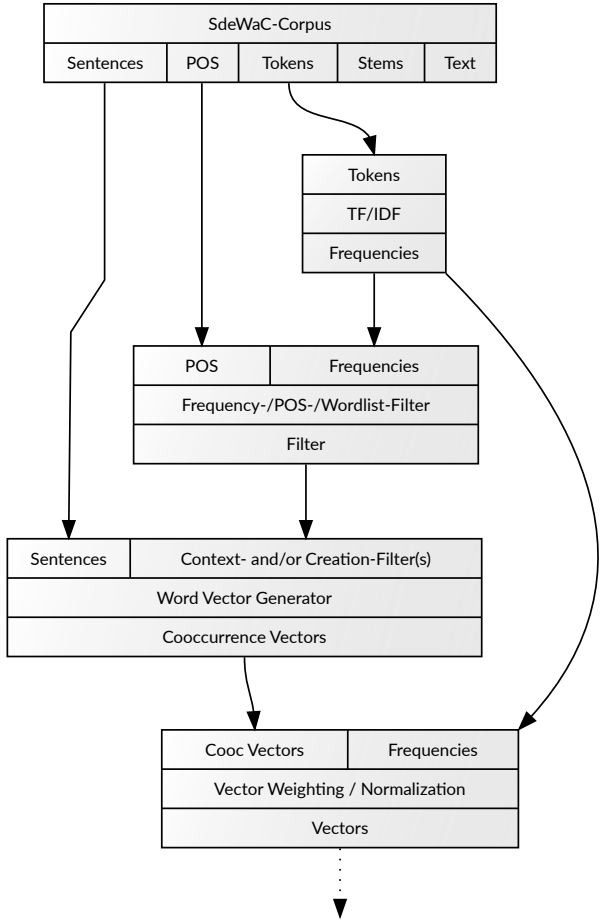


Abbildung 7.1: Schematischer Aufbau zur Erstellung von Wortvektoren in Tesla. Die resultierenden Vektoren können über die in die VectorWeighting-Komponente eingebaute Plotting-Funktionalität visualisiert werden.

Zu Kontrollzwecken können die erzeugten Vektoren über die integrierte Plotting-Funktionalität visualisiert werden. Hierbei wird der Vektorraum mittels multidimensionaler Skalierung auf ein zweidimensionales Raster reduziert (siehe Abschnitt 6.2.8). Anhand der Plots kann einfacher nachvollzogen werden, inwieweit sich die Parameter auf die Art der Ähnlichkeit auswirken. Aus Gründen der Lesbarkeit kann, anstatt den gesamten Vektorraum zu projizieren, auch nur eine Teilmenge geplottet werden, deren Größe über die Komponentenkonfiguration festgelegt wird.

7.1.2 Parametrisierung

Wie in Kapitel 6 beschrieben, gibt es eine Vielzahl von Parametern und Faktoren, die Einfluss auf die Art und Qualität der Vektoren haben. Bei der einfachsten Form der Erstellung von Kookkurrenzvektoren richtet sich die Anzahl der Merkmale (und damit die Vektorlänge) nach der Anzahl der verschiedenen Wortformen im Korpus. Im Fall des LCC-Korpus mit 1 Million Sätzen sind dies insgesamt 696.014 verschiedene Wortformen.[117] Vektoren dieser Länge sind nicht nur unhandlich, sie enthalten auch nur zu einem gewissen Grade brauchbare Informationen. Beim Vergleich der Vektoren fallen niederfrequente Merkmale allgemein weniger stark ins Gewicht.[118] So konnten Levy/Bullinaria (2001) anhand von Vergleichstests zeigen, dass es ausreicht, die häufigsten Types als Merkmale zu nutzen. Mit dieser sehr einfachen Heuristik lässt sich die Vektorlänge stark begrenzen, ohne dass komplexere Methoden eingesetzt werden müssten.[119]

Die Verkürzung der Vektoren ist von entscheidender Bedeutung für die nachfolgenden Verarbeitungsschritte: Für die Experimente in diesem Kapitel ist es wichtig, mit möglichst kurzen Vektoren zu arbeiten, da in der weiteren Verarbeitung eine Vielzahl von rechenintensiven Ähnlichkeitsvergleichen vorgesehen ist, etwa bei der Ermittlung der ähnlichsten Wörter, vor allem aber bei den in Abschnitt 7.3 eingesetzten Clusteranalysen. Zum einen fallen bei einer geringeren Zahl von Merkmalen weniger Vergleichsoperationen an, zum anderen sind die Ähnlichkeitsvergleiche umso zuverlässiger, je mehr Merkmale in den Vektoren belegt sind, da die Ähnlichkeitsberechnung damit ein robustes Fundament hat.[120]

Dennoch dürfen die Vektoren nicht beliebig kurz sein: Während eine Konzentration nur auf wenige sehr hochfrequente Merkmale eine Betonung der Kookkurrenz mit Funktionswörtern (sogenannte geschlossene Klassen) und damit eine stärker funktional-grammatische Ausprägung bedeutet, bieten längere Vektoren ein deutlich differenzierteres Bild hinsichtlich der semantischen Ähnlichkeiten, da bei einer größeren Zahl von Merkmalen eine feinere Differenzierung im Verwendungsmuster möglich ist. Aus diesen Überlegungen ergeben sich die folgenden, in Tabelle 7.1 zusammengefassten Komponenten-Konfigurationen für die Erstellung von Kookkurrenzvektoren.

117 Bei einer entsprechend einfachen Tokenisierung sind hier jedoch auch Zahlen sowie Satz- und Sonderzeichen enthalten.

118 Insbesondere sind Wörter, die nur ein Mal auftreten, als Merkmal nicht aussagekräftig: Da sie beim Vektorvergleich keine Rolle spielen, bieten sie keinerlei Mehrwert für die Repräsentation und können weggelassen werden. Allein dadurch reduziert sich die Anzahl der Merkmale auf ca. die Hälfte.

119 Die Entscheidung für die Heuristik von Levy & Bullinaria (2001) folgt demnach dem Ökonomieprinzip, oftmals auch als »Ockhams Rasiermesser« bezeichnet: Es wird das einfachste verfügbare Vorgehen gewählt, da es nicht erkennbar schlechter ist als andere, aufwändigere Verfahren, wie etwa die Merkmalsauswahl anhand der Varianz (vgl. Lund/Burgess 1996) oder eine nachträgliche Dimensionsreduktion mittels Singular Value Decomposition (vgl. Landauer/Dumais 1997).

120 Bei niederfrequenten Merkmalen ist der Wert in den allermeisten Fällen 0, so dass die entsprechenden Merkmale als nicht diskriminierend anzusehen sind.

Kookkurenzvektoren		
Korpus	SdeWaC, 1 Mio. Sätze, deutsch	
	Stemming	ja
Context Filter (Merkmalsauswahl)	Frequency Filter	
	Range	8.000 häufigste
Creation Filter (Anzahl Vektoren)	Frequency Filter	
	Range	100–10.100 häufigste (ohne häufigste 100)
Vektoren	Fenster	3
	Nachbarschaft (HAL)	nein (keine Gewichtung)
	Vektorlänge	8.000
	Anzahl Vektoren	10.000
Normalisierung	Euklidische Länge	
Gewichtung	Pointwise Mutual Information (PMI)	

Tabelle 7.1: Konfiguration der beteiligten Komponenten im Experiment zur Erstellung von Kookkurenzvektoren in Tesla.

Um auch bei den hier auf 8.000 Elemente verkürzten Vektoren einen möglichst hohen Informationsgehalt zu erreichen, werden anstelle der Vollformen die Wortstämme betrachtet. Durch das Stemming können jeweils mehrere Wortformen im Sinne einer Äquivalenzklasse zu einem gemeinsamen Merkmal zusammengefasst werden, was zu einer höheren Zahl von möglichen Kookkurrenzen führt. Die Anzahl der Kookkurrenzen wird zusätzlich erhöht, indem ein leicht vergrößertes Kontextfenster eingesetzt wird, anstatt nur die direkten Nachbarn zu berücksichtigen.[121] Im Vorgriff auf die weiteren Experimente wird zudem die Anzahl der zu erstellenden Vektoren über den CreationFilter auf 10.000 begrenzt, um dadurch in weiteren Verarbeitungsschritten die Anzahl der notwendigen Vergleichsoperationen einzuschränken.[122]

121 Wie in Abschnitt 4.1.2 beschrieben, werden damit neben den lokalen, eher grammatisch orientierten Beziehungen, die ein Wort zu seinem Umfeld unterhält, auch stärker thematisch orientierte Relationen mit einbezogen.

122 Die Wahl einer sinnvollen Anzahl hängt dabei auch von der Länge der zu vergleichenden Vektoren ab: Bei der hier gewählten Vektorlänge von 8.000 Merkmalen und einer Gesamtzahl von 10.000 Vektoren sind insgesamt bereits 80 Millionen Vergleichsoperationen nötig; wird die Anzahl der Vektoren oder deren Länge erhöht, führt dies schnell zu einer sehr langen Verarbeitungsdauer.

7.1.3 Referenzräume

Wesentliches Ziel der Experimente in diesem Kapitel ist eine Simulation des Prozesses der Bedeutungskonstitution. Gemäß der Vorhersage auf Grundlage des Modells geht mit den unterschiedlichen Kontextualisierungen eines Wortes in der Regel eine Veränderung der Repräsentation einher. Angesichts der mit dem Vektorraum gegebenen Möglichkeit zur räumlichen Darstellung kann diese Veränderung als eine Art ›Bewegung‹ innerhalb des Vektorraums angesehen werden. Um diese Bewegung besser nachvollziehbar zu machen, bedarf es neben dem Ausgangsvektor einer geeigneten Bezugsgröße. Hier bietet es sich an, eine Reihe zusätzlicher Referenzpunkte im Wortraum anzunehmen, gegenüber denen die Abweichung dargestellt werden kann. Abb. 7.2 zeigt eine entsprechende Erweiterung des Workflows aus Abschnitt 7.1.1, bei der eine eigene Komponente für die Erstellung eines solchen Referenzraums hinzukommt.

Die Parametrisierung der Wortvektoren erfolgt hier gemäß den Angaben im vergangenen Abschnitt (siehe Tabelle 7.1). Mit der zusätzlich eingesetzten

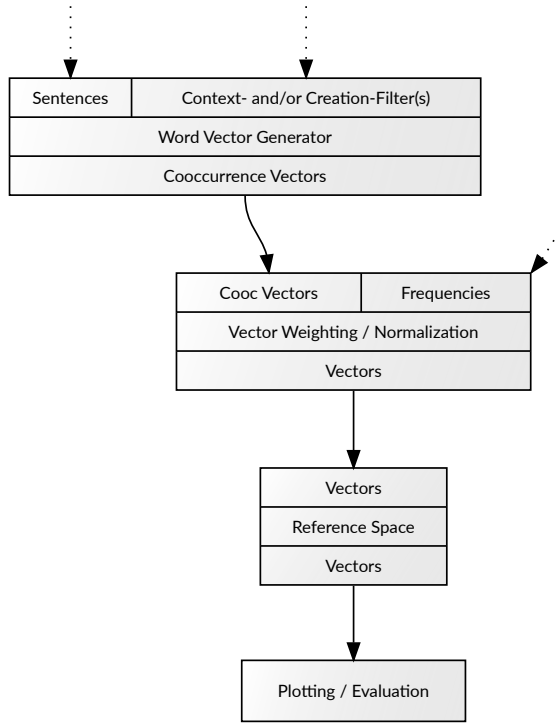


Abbildung 7.2: Schematischer Aufbau zur Erstellung von Referenzräumen auf Grundlage von Wortvektoren in Tesla.

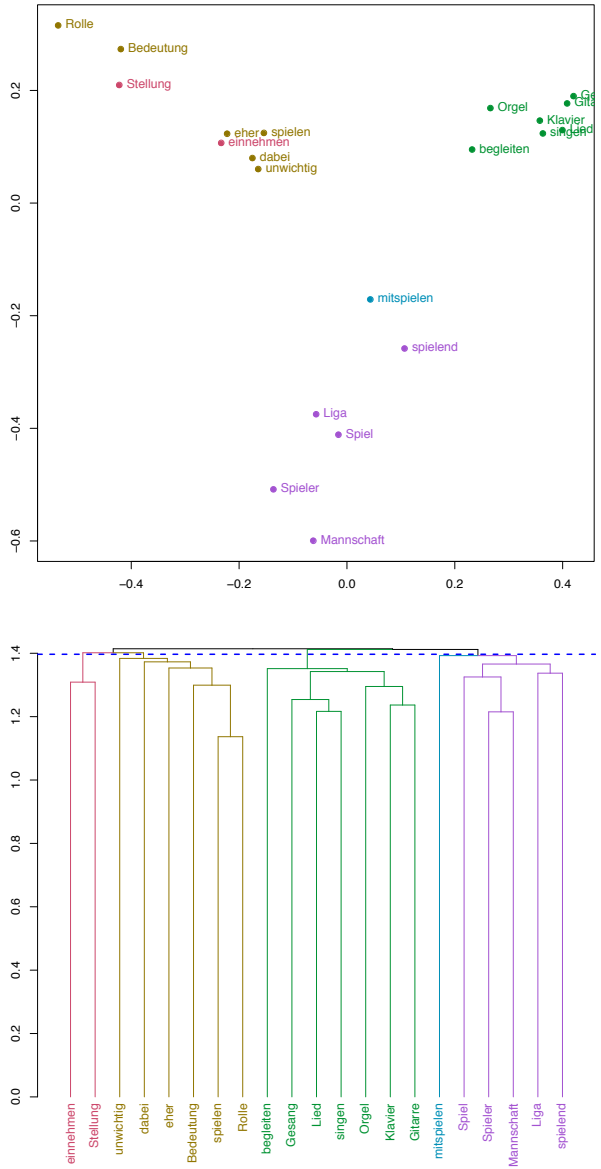


Abbildung 7.3: Beispiel für einen Referenzraum. Ausgehend von dem Zielwort *spielen* werden die 20 ähnlichsten Wörter ermittelt; daraus ergibt sich ein Ausschnitt des Wortraums, der nur das nähere Umfeld des Zielworts umfasst. Bei einer zusätzlichen Anwendung eines hierarchischen Clusterings finden sich mindestens drei größere Gruppen, was bereits einen deutlichen Hinweis gibt auf die verschiedenen, im Zielwort *spielen* enthaltenen Bedeutungsmöglichkeiten.

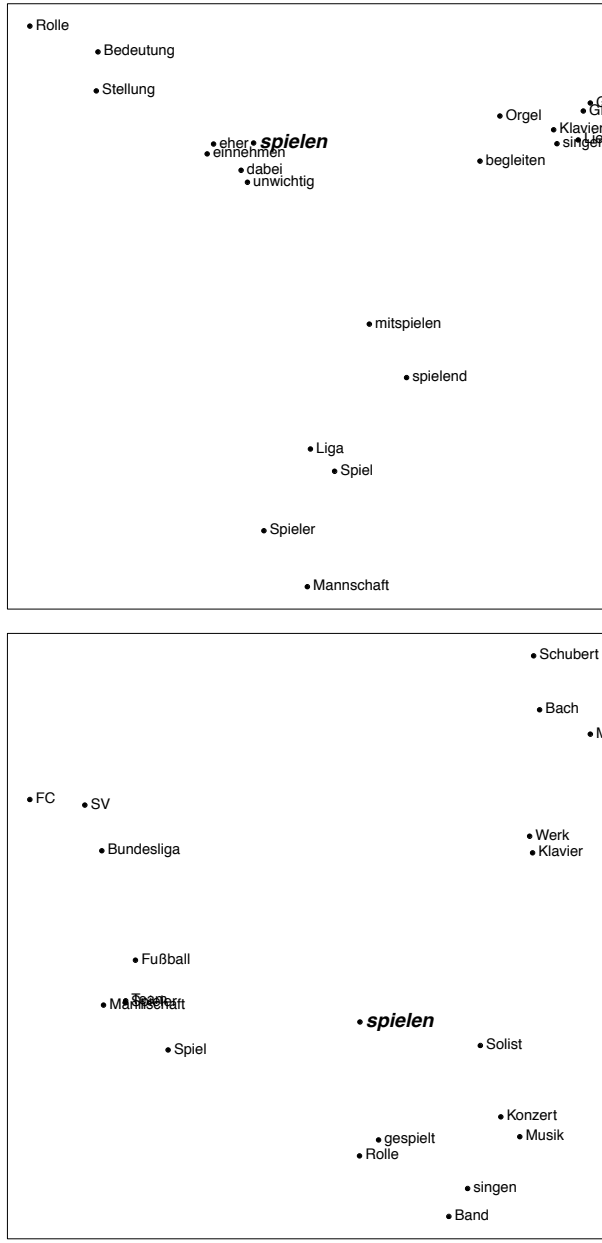
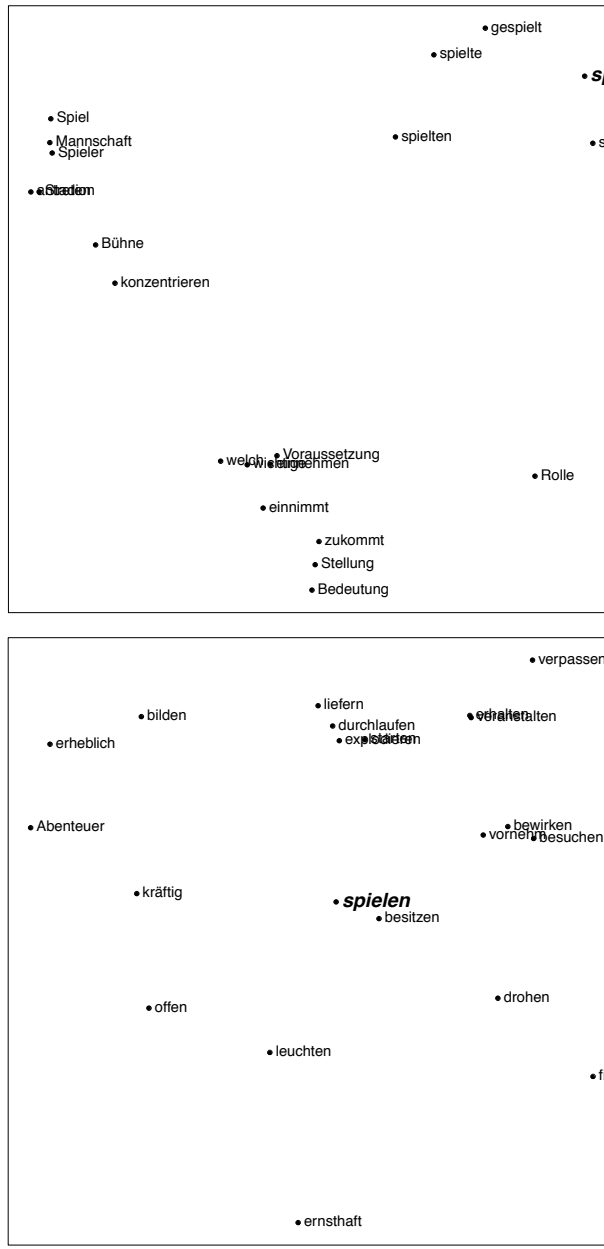


Abbildung 7.4: Auswirkungen unterschiedlicher Parametrisierungen und Datengrundlagen auf die Beschaffenheit des Referenzraums. Oben: Gewichtete Wortstämme im SdeWaC-Korpus (entspricht Abb. 7.3). Unten: Gewichtete Wortformen (SdeWaC).



Zu Abbildung 7.4: Oben: Gewichtete Wortstämme im LCC-Korpus. Unten: Ungewichtete Wortstämme (SdeWaC).

ReferenceSpace-Komponente wird zum jeweiligen Zielwort eine feste Anzahl der ähnlichsten Wörter ermittelt und nur diese Teilmenge erscheint im Plot. In den Plots wird der Vektorraum somit jeweils aus Perspektive eines einzelnen Wortes dargestellt, indem nur dessen nähere Umgebung gezeigt wird. Um auch bei einer ausschnittsweisen Betrachtung ein möglichst repräsentatives Bild des Vektorraums zu erhalten, ist die Verfügbarkeit von möglichst vielen Vergleichsvektoren Voraussetzung. Die Anzahl wird über den CreationFilter gesteuert und wurde im Experiment auf 10.000 begrenzt.

Abb. 7.3 zeigt einen solchen Referenzraum für das Beispielwort *spielen*, bestehend aus dem Zielwort und seinen 20 ähnlichsten Wörtern. Mithilfe der Plotting-Funktionalität kann zusätzlich ein hierarchisches Clustering vorgenommen werden, um die Struktur des Referenzraums zu verdeutlichen. Grundlage für die Ähnlichkeitsberechnung sind die jeweiligen Verwendungsmuster der Wörter. Unter den ähnlichsten Elementen finden sich daher zum einen Wörter, die auffällig häufig im Kontext des Zielworts verwendet werden (etwa *Fußball, Klavier* etc.), zum anderen auch solche, die in anderen, ähnlichen Kontexten verwendet werden (etwa *singen*, *mitspielen* etc.). Bereits hier zeigt sich das mehrdeutige Potential des Beispielworts, insofern sich unter den abgebildeten Elementen recht deutlich verschiedene thematische Gruppen ausmachen lassen.

Welche Wörter letztlich mit im Plot erscheinen, ist dabei unmittelbar davon abhängig, wie zuvor die Kookkurrenzvektoren errechnet werden und auf welcher Datengrundlage dies erfolgt. Beides hat direkte Auswirkungen darauf, wie die jeweiligen Verwendungsmuster beschaffen sind, und damit auch auf die Ähnlichkeiten zwischen den Elementen, die durch die Muster repräsentiert werden. Dementsprechend anders stellt sich der Referenzraum für das Zielwort *spielen* dar, wenn eine andere Berechnungsgrundlage gewählt wird, wie die Plots in Abb. 7.4 verdeutlichen.

In den Experimenten zur Bedeutungskonstitution, die im Folgenden beschrieben werden, werden die Referenzräume gemäß der Parametrisierung in Tabelle 7.1 erstellt. Dort dienen sie als eine Art Referenzrahmen, um die mit der Bedeutungskonstitution verbundene Transformation der Vektoren nachvollziehbar zu machen.

7.2 Bedeutungskonstitution in Einzelkontexten

In diesem Abschnitt wird die konkrete Umsetzung des Prozesses der Bedeutungskonstitution beschrieben. Das Vorgehen im Experiment richtet sich dabei nach der in Abschnitt 5.2 formulierten Modellierung, der zufolge die Bedeutungskonstitution als kontextuelle Aktivierung im Vektorraum verstanden werden kann. Mit den im vergangenen Abschnitt beschriebenen Referenzräumen auf Grundlage von Kookkurrenzvektoren steht nun zudem eine Zielstruktur zur Verfügung, in der die hierbei angenommene Bedeutungsveränderung durch den Kontext

visualisiert werden kann. Hierbei wird das mit dem Vektorraum verbundene Prinzip der geometrischen Metapher direkt ausgenutzt: indem die Abweichung gegenüber dem ursprünglichen Vektor über die Veränderung der Position im Vektorraum dargestellt wird, kann sie in gewissem Sinne als ›Bewegung‹ interpretiert werden, die den Prozess der Bedeutungskonstitution simuliert.

In Abschnitt 7.2.1 wird zunächst der grundlegende Workflow des Experiments beschrieben. Im Zuge des Experiments werden für ausgewählte Einzelvorkommen verschiedener Beispielwörter unter Hinzunahme der jeweiligen Kontexte lokale Repräsentationen erstellt und in den zuvor beschriebenen Referenzraum projiziert. In Abschnitt 7.2.2 werden die Konfigurationsmöglichkeiten der beteiligten Komponenten beschrieben und die konkrete Parametrisierung bei der Durchführung der Beispielanalysen erläutert, deren Ergebnisse anschließend in Abschnitt 7.2.3 diskutiert werden.

7.2.1 Aufbau des Experiments

Als Eingabeinformation für den Prozess der Bedeutungskonstitution dienen hier die im vergangenen Abschnitt beschriebenen Kookkurrenzvektoren. Leitgedanke des Experiments ist eine kontextuelle Aktivierung von (Teil-)Bedeutungen durch den Kontext: der initiale Kookkurrenzvektor enthält zunächst die Gesamtheit der möglichen Verwendungsweisen des repräsentierten Wortes, und erst durch die Hinzunahme des jeweiligen Kontextes wird eine konkrete Bedeutung aktiviert. Der Prozess wird umgesetzt, indem für jedes Einzelvorkommen ein eigener Vektor erstellt wird: hierbei wird der Vektor eines gegebenen Zielworts mit den Vektoren der Kontextelemente kombiniert. Die resultierenden Vektoren können im Anschluss in einen Referenzraum projiziert werden, um die Veränderung gegenüber dem Zielwort zu visualisieren. Abb. 7.5 zeigt den schematischen Workflow des entsprechenden Tesla-Experiments. Herzstück des Versuchsaufbaus ist die Komponente zur Erstellung von Repräsentationen für Einzelkontexte, in der Abbildung als Local Context Vectors bezeichnet.

Auf Grundlage der in Abschnitt 5.2 angestellten Überlegungen wurden im Zuge dieser Arbeit zwei unterschiedliche Varianten realisiert: Während der Ursprungsvektor in der Umsetzung als ContextVectors-Komponente direkt mit dem Zentroid einer festgelegten Anzahl von Kontextelementen kombiniert wird,[123] wird in der CollocationVector-Komponente in einem zusätzlichen Schritt zunächst noch

123 Die Implementation der Komponente erfolgte damit in enger Anlehnung an das in Schütze (1998) beschriebene Vorgehen (vgl. dazu auch Abschnitt 5.2.1). Anders als bei Schütze kann der zu berücksichtigende Kontext hier über ein variables Kontextfenster definiert werden, das mindestens die direkten Nachbarn und maximal einen vollständigen Satz umfasst (bei Schütze wird dagegen ein längerer, satzübergreifender Kontext eingesetzt, was hier aufgrund der Beschaffenheit der Korpora nicht möglich ist, da diese keine fortlaufenden Sätze enthalten).

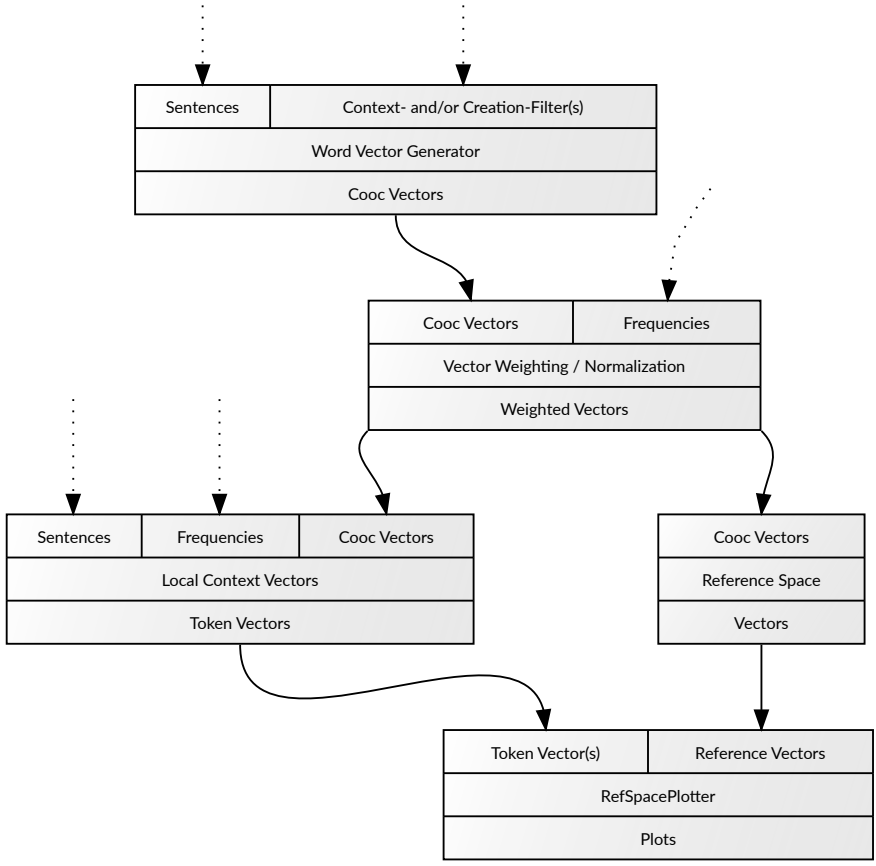


Abbildung 7.5: Versuchsaufbau zur Analyse der Bedeutungskonstitution in ausgewählten Kontexten. Die in der Komponente Local Context Vectors für die Einzelvorkommen erzeugten Vektoren werden gemeinsam mit den durch die Reference Space-Komponente ausgewählten Vergleichsvektoren visualisiert.

eine Gewichtung der Kontexte vorgenommen, so dass nur ausgewählte Kontextelemente in den Prozess mit einbezogen werden.

In beiden Varianten kann wahlweise eine festgelegte Anzahl von Kontexten verarbeitet oder alternativ eine Liste von Sätzen bzw. Teilsätzen oder Phrasen eingelesen werden, welche die Auswahl geeigneter Kontexte regeln. Dadurch können ganz gezielt bestimmte Kontexte einzeln oder gemeinsam betrachtet werden. Die Vektoren der Einzelkontexte werden im Anschluss zusammen mit den Vektoren des separat erstellten Referenzraums an die Komponente RefSpacePlotter weitergereicht und gemeinsam visualisiert. Als Ergebnis werden verschiedene Plots ausgegeben, in denen die Veränderung gegenüber dem jeweils betrachteten

Ausgangsvektor nachvollzogen werden kann. Die Experimentbeschreibung konzentriert sich im Folgenden auf die CollocationVectors-Komponente.

7.2.2 Parametrisierung

Im zentralen Verarbeitungsschritt des Experiments wird im Wesentlichen das von Schütze (1998) beschriebene Prinzip der Kombination von Zielwortvektor und Kontext umgesetzt, jedoch werden hierbei nicht alle Kontextwörter mit einbezogen. Hintergrund ist die Annahme, dass nicht alle Elemente im Kontext in gleichem Maße an der Bedeutungskonstitution beteiligt sind (siehe Abschnitt 5.2.2); der Name der Komponente verweist dabei auf das zugrunde gelegte Konzept der Kollokation, das hier im Sinne einer nicht zufälligen Wortverbindung verstanden wird. Um dies in der Umsetzung angemessen zu berücksichtigen, wird in der CollocationVectors-Komponente zunächst eine Bewertung der Kontextwörter aus Perspektive des Zielworts vorgenommen, bei der die Signifikanz des gemeinsamen Auftretens berechnet wird: nur die signifikantesten Elemente werden anschließend für die Erstellung des Kontextvektors verwendet.

Zentrale Parameter der CollocationVector-Komponente sind die Anzahl der zugelassenen Kontextelemente sowie das Assoziationsmaß, mit dem die signifikantesten Elemente ermittelt werden. Für die Berechnung der Signifikanz kann über die Komponentenkonfiguration zwischen einer Reihe von Maßen gewählt werden (siehe Abschnitt 6.2.6).[124] Zusätzlich kann eine Positionsgewichtung mit einbezogen werden, bei der Wörter mit größerem Abstand zum Zielwort ein geringeres Gewicht erhalten als die Elemente im näheren Umfeld, damit weiter entfernt liegende potentielle Kollokate nicht fälschlich zu hoch bewertet werden. Die konkrete Parametrisierung der im Experiment beteiligten Verfahrensbestandteile ist in Tabelle 7.2 zusammengefasst.

Die Beispielanalysen im nachfolgenden Abschnitt werden unter Verwendung der Log-Likelihood-Ratio (LLR) mit einer zusätzlichen Positionsgewichtung durchgeführt. Für die Erstellung der Kontextvektoren wird dabei immer nur das signifikanteste Kontextelement berücksichtigt. Weil die Werte der LLR relativ stark streuen, werden sie in der Komponente normalisiert, um den Wertebereich zwischen 0 und 1 zu fixieren. In der hier gewählten Konfiguration werden die errechneten Signifikanzwerte durch eine Folge von festen Werten ersetzt, die ausgehend von dem Wert 1 für das signifikanteste Element in jedem Schritt um die Hälfte abnehmen (das heißt 0,5 für das zweitsignifikanteste Element, 0,25 für

124 Die Kookkurrenzwerte, die für die Berechnung der Assoziationsmaße benötigt werden, werden separat berechnet, um sie bei einer mehrfachen Durchführung des Experiments mit veränderten Parametern wiederverwenden zu können. Hierfür wurde eine eigene CoocHelper-Komponente implementiert, die hier jedoch nicht mit in die Workflow-Grafik aufgenommen wurde (siehe dazu Anhang A).

Einzelvorkommen: Kollokationen		
Korpus	SdeWaC, 1 Mio. Sätze, deutsch	
	Stemming	ja
Vektoren	Fenster	3
	Länge	8.000 (häufigste)
	Anzahl	10.000 (ohne 100 häufigste)
	Gewichtung	PMI, normalisiert
CollocationVectors	Kontextelemente	1
	Assoziationsmaß	Log-Likelihood
	Positionsgewichtung	ja
	Normalisierung	Fester Wertebereich
Referenzraum	20 ähnlichste + Zielwort	

Tabelle 7.2: Parametrisierung der wesentlichen Verfahrensbestandteile im Experiment zur Bedeutungskonstitution in Einzelkontexten. Die Parameter der Kookkurrenzvektoren sowie des Referenzraums wurden aus Abschnitt 7.1 übernommen; die der CollocationVectors-Komponente werden im Text erläutert.

das dritte etc.).[125] In den Beispielanalysen wird die Signifikanz demnach in jedem Kontext gleich bewertet, wobei der Wert 1 bedeutet, dass Zielwort und Kollokat beim Zusammenführen der Vektoren gleich stark gewichtet werden.

Neben den hier angegebenen Parametern bietet die Komponente noch weitere Konfigurationsmöglichkeiten, die jedoch in diesem Abschnitt nicht zum Einsatz kommen und deshalb nicht mit in die Tabelle aufgenommen wurden. So kann unter anderem ein Schwellwert angegeben werden, unterhalb dessen die Wörter als nicht signifikant angesehen werden, des Weiteren kann der errechnete Wert durch Angabe eines zusätzlichen Faktors global verstärkt oder abgeschwächt werden.

7.2.3 Beispielanalysen

Die Abbildungen 7.6 und 7.7 zeigen die Ergebnisse des Experiments für ausgewählte Kontexte des Zielworts *spielen* unter Verwendung der CollocationVectors-Komponente. In jedem der abgebildeten Kontexte erfolgt eine leichte Veränderung der Repräsentation, die sich in einer sichtbaren Veränderung der Position äußert.

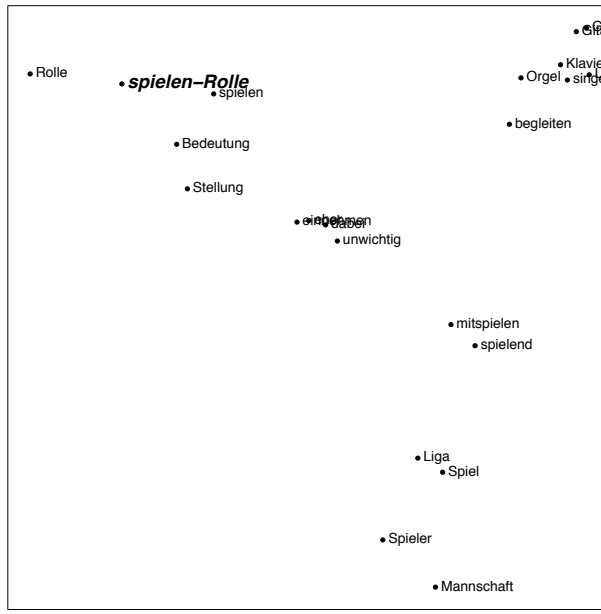
125 Alternativ werden die Werte anhand des errechneten Maximalwerts normalisiert, so dass sie ebenfalls zwischen 0 und 1 liegen, wobei die Streuung in Bezug auf das Verhältnis der Werte untereinander erhalten bleibt. Beide Formen der Normalisierung sind für sämtliche Signifikanzmaße anwendbar.

Die Plots legen das zugrunde gelegte Prinzip der kontextuellen Aktivierung durch Kollokate offen: Sind deren Vektoren mit im Referenzraum enthalten, wird der Kontextvektor zwischen Zielwort und Kollokat projiziert (siehe *Rolle* und *Spiel*, linke Seite), andernfalls orientiert er sich in die Nähe thematisch ähnlicher Wörter (siehe *Fußball* im Vergleich zu *Spiel* in den unteren Plots). Ähnliche Kontextvektoren werden demnach stets in ähnliche Bereiche projiziert, wie die Beispiele in Abbildung 7.7 verdeutlichen. Auch hier verändert sich die Position des lokal erzeugten Vektors für jeden Kontext. Ohne dass die Vektoren der Kollokate selbst im Referenzraum enthalten sind (hier *Musik, Melodie* und *Konzert*), werden die Kontextvektoren aufgrund der Ähnlichkeit der enthaltenen Kollokate stets in die gleiche Region projiziert. Noch deutlicher wird dies, wenn man die betreffenden Kontexte gemeinsam betrachtet; Abbildung 7.8 zeigt eine solche Mehrfachprojektion für verschiedene Beispielwörter.

In der gemeinsamen Projektion mehrerer Kontexte wird zum einen erneut die Veränderung der einzelnen Verwendungen gegenüber dem Zielwort deutlich, zum anderen zeigt sich hier, dass die einzelnen Kontextualisierungen auch untereinander verschieden starke Affinitäten aufweisen. Aus den Beispielen in Abbildung 7.8 wird zudem ersichtlich, dass sich der Prozess für Wörter verschiedener Wortarten gleichermaßen auswirkt. Die Differenzierung scheint dabei für Verben (*spielen*) und Adjektive (*scharf*) deutlicher auszufallen als für Nomen (*Krone*) oder kategorial ambige Wörter (*erwachsen*, das als Adjektiv oder als Verb verwendet werden kann). Dass die Veränderungen nicht in allen Plots gleichermaßen klar sichtbar werden, liegt hier jedoch vor allem an der Beschränkung der Perspektive durch die Beschaffenheit der jeweiligen Referenzräume, insofern diese nur einen sehr kleinen Ausschnitt des Gesamtdatenraums zeigen. So ist die Veränderung immer dann besonders gut erkennbar, wenn der Vektor des jeweiligen Kollokats im Plot enthalten ist, so etwa im Falle von *scharf-Gegner* in Abbildung 7.8 oder auch *spielen-Klavier* in Abbildung 7.7 – bei einer größeren Anzahl von Referenzpunkten würde dies entsprechend häufiger eintreten.[126] Der entscheidende Punkt ist hier jedoch ein anderer: Auch wenn die Differenzierung in den jeweiligen Referenzräumen nicht immer klar erkennbar ist, so findet sie in Bezug auf den Gesamtdatenraum dennoch in jedem Kontext statt, da dieser im Gegensatz zum Referenzraum vollständig ist und demnach auch die nicht mit abgebildeten Kollokate enthält.

Die wesentliche Gemeinsamkeit der hier gezeigten Beispiele besteht somit darin, dass durch die Hinzunahme des Kontextes in jeder Verwendung eine Veränderung der Repräsentation erfolgt, die mit einer veränderten Position im Vektorraum einhergeht. Interpretiert man diesen räumlichen Unterschied nun auf Grundlage der

126 Ein anderes Bild würde sich auch dann ergeben, wenn die zugrunde gelegten Vektoren in einer anderen Konfiguration oder auf Grundlage eines anderen Korpus eingesetzt werden, da dies großen Einfluss auf die Beschaffenheit der jeweiligen Referenzräume hat (vgl. dazu Abb. 7.4 in Abschnitt 7.1.3). Der Prozess als solcher bleibt davon jedoch unberührt: Auch hier ›bewegt‹ sich die lokal erzeugte Repräsentation in Richtung des jeweiligen Kollokats.



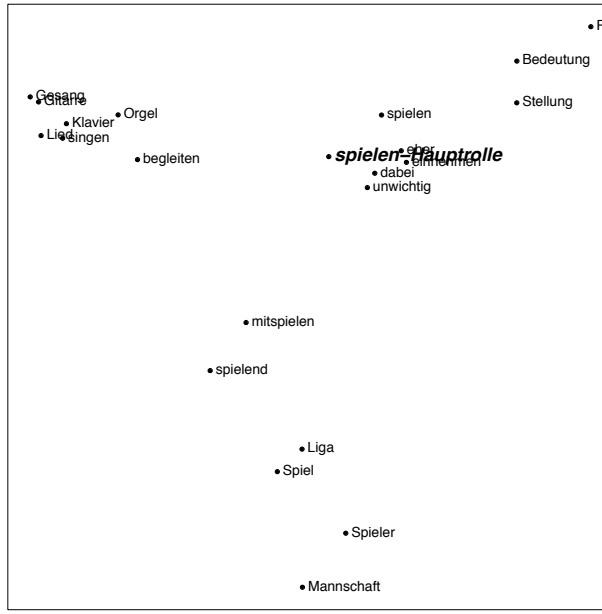
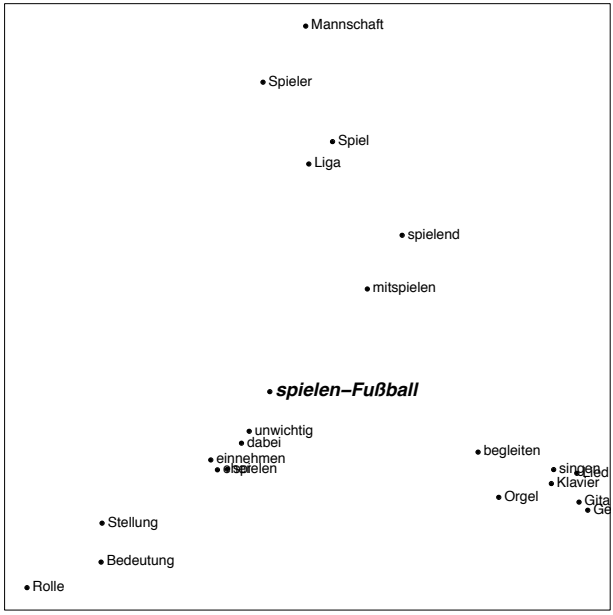


Abbildung 7.6: Projektion ausgewählter Kontexte in den Referenzraum des Zielworts *spielen*. Der Einfluss der Kollokate ist deutlich erkennbar: Ausgehend vom Zielwortvektor werden die Kontextvektoren in jedem Kontext in Richtung thematisch ähnlicher Wörter ›gezogen‹.


Spiel
Mannschaft
Spieler
Liga
spielen–Spiel
spielend
mitspielen
spielen
begleiten
singen
Lied
Gitarre
Klavier
Gesang
Orgel
dabei
unwichtig
einnehmen
Stellung
Bedeutung


Mannschaft
Spieler
Spiel
Liga
spielend
mitspielen
spielen–Fußball
unwichtig
dabei
einnehmen
begleiten
Orgel
Klavier
Stellung
Bedeutung
Rolle

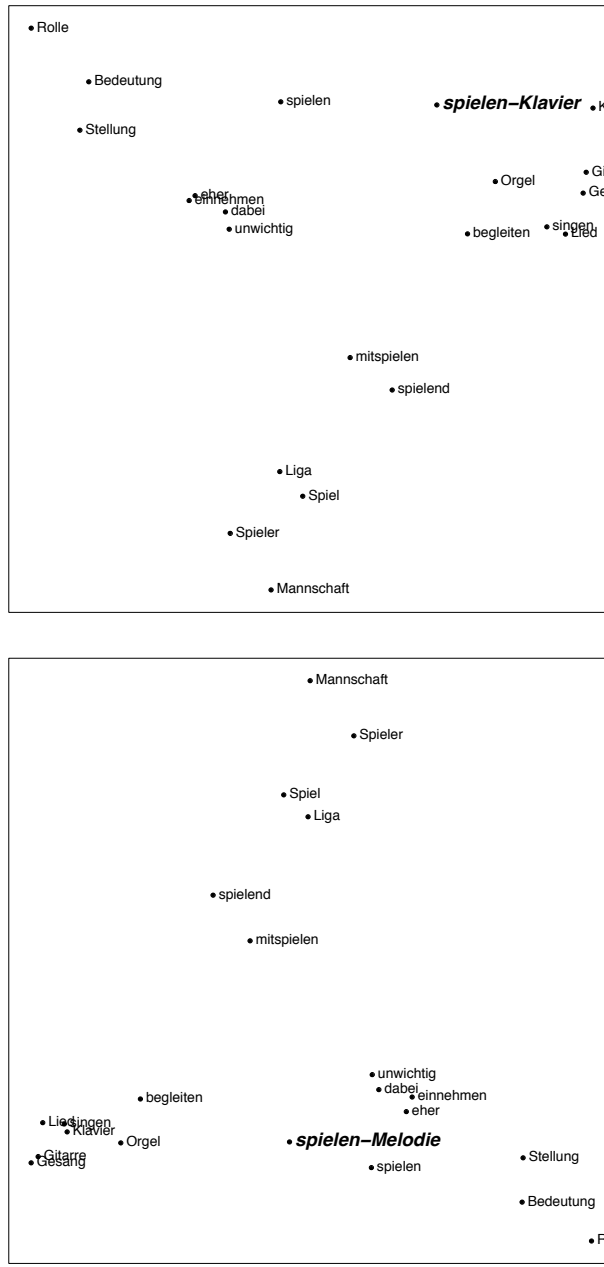
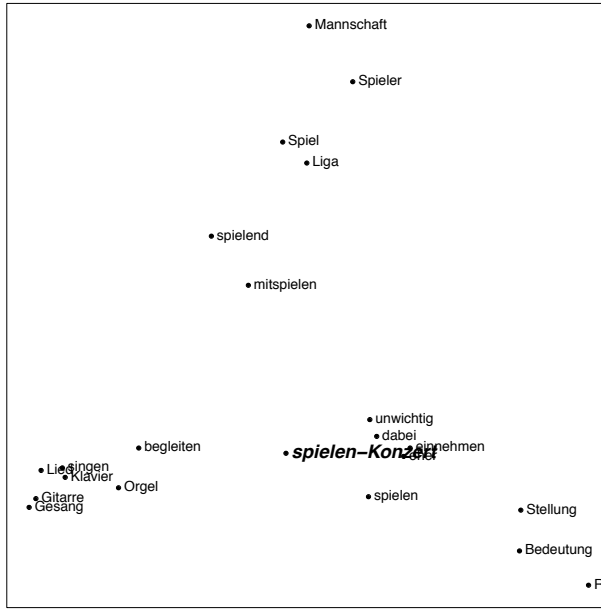


Abbildung 7.7: Projektion thematisch ähnlicher Kontexte in den Referenzraum des Zielworts *spielen*. Aufgrund der Ähnlichkeit der Kollokate werden die Kontextvektoren hier ebenfalls in einem ähnlichen Bereich positioniert. Die Vektoren der Kollokate müssen dabei selbst nicht mit im Referenzraum enthalten sein.


Mannschaft
Spieler
Spiel
Liga
spielend
mitspielen
unwichtig
dabei
einnehmen
eher
begleiten
singen
Lied
Klavier
Orgel
Gitarre
Gesang
spielen–Musik
spielen
Stellung
Bedeutung
Ro


Mannschaft
Spieler
Spiel
Liga
spielend
mitspielen
unwichtig
dabei
einnehmen
eher
begleiten
singen
Lied
Klavier
Orgel
Gitarre
Gesang
spielen–Konzert
spielen
Stellung
Bedeutung
Ro

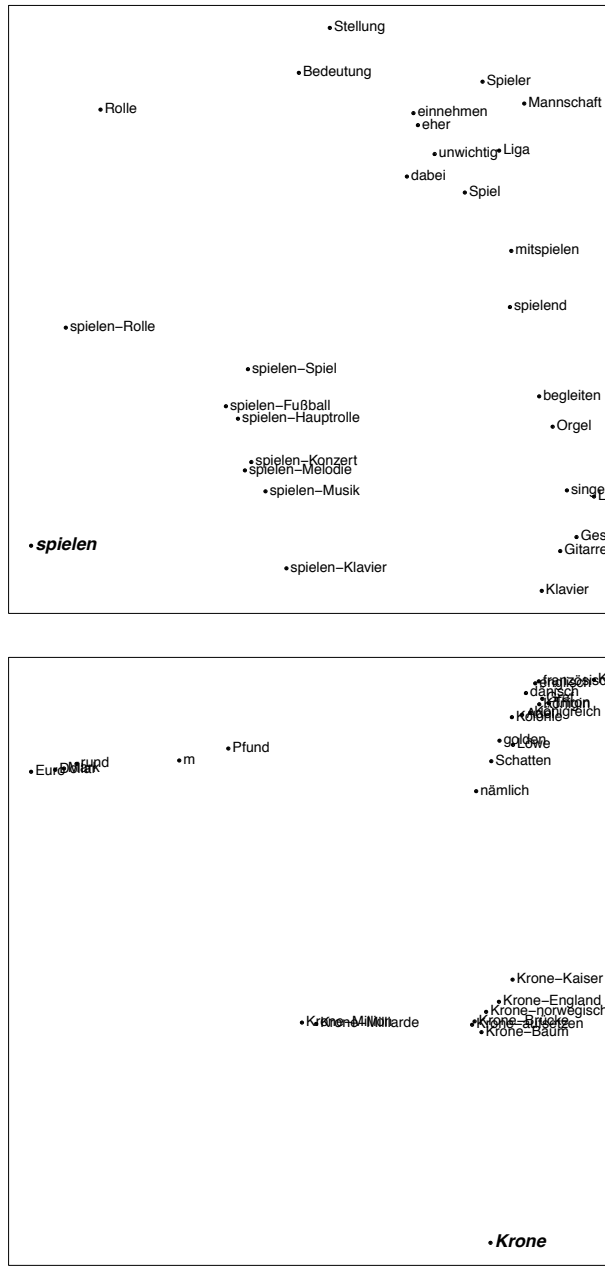
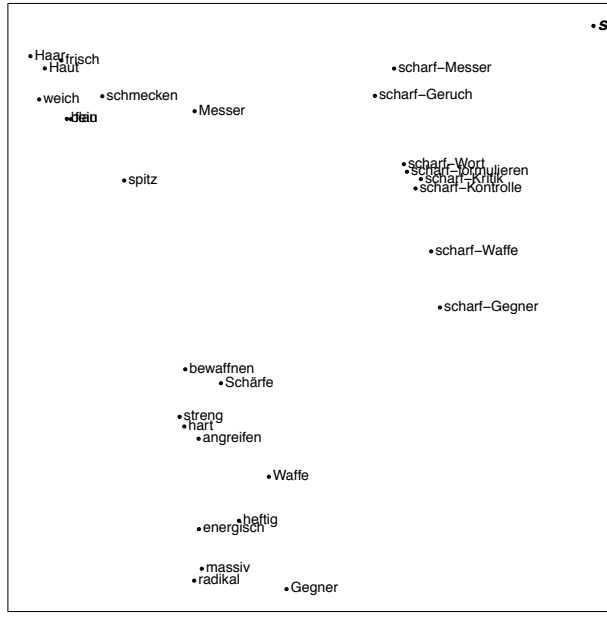
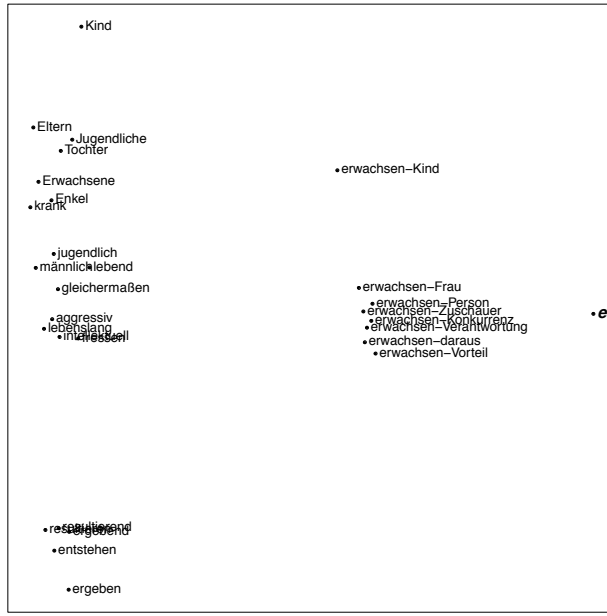


Abbildung 7.8: Mehrfachprojektion ausgewählter Kontexte für verschiedene Wörter. Neben der Veränderung gegenüber dem Zielwort werden hier auch die Unterschiede zwischen den einzelnen Kontextvektoren deutlich: ähnliche Kontextvektoren orientieren sich in eine ähnliche Richtung.


sc
Haar
frisch
Haut
scharf–Messer
weich
schmecken
scharf–Geruch
Messer
spitz
scharf–Wort
scharf–formulieren
scharf–Kritik
scharf–Kontrolle
scharf–Waffe
scharf–Gegner
bewaffnen
Schärfe
streng
hart
angreifen
Waffe
heftig
energisch
massiv
radikal
Gegner


Kind
Eltern
Jugendliche
Tochter
Erwachsene
Enkel
krank
erwachsen–Kind
jugendlich
männlich
lebend
gleichermaßen
erwachsen–Frau
erwachsen–Person
erwachsen–Zuschauer
aggressiv
lebenslang
erwachsen–Konkurrenz
erwachsen–Verantwortung
erwachsen–daraus
erwachsen–Vorteil
er
entstehen
ergeben

geometrischen Metapher als semantischen Unterschied, so lässt sich hieraus eine kontextbedingte Bedeutungsvariation ablesen. Damit greift hier auch die distributionelle Hypothese, insofern das veränderte Verwendungsmuster als Veränderung der Bedeutung gedeutet wird. Vor dem Hintergrund der in dieser Arbeit getroffenen theoretischen Vorannahmen ist die Variation dabei als das Resultat der jeweiligen Bedeutungskonstitution anzusehen. Abhängig davon, welche Wörter im Kontext auftreten, wird der Ursprungsvektor in die entsprechende Richtung gelenkt. Die konkrete Bedeutung konstituiert sich somit erst lokal durch die kontextuelle Aktivierung einer der implizit enthaltenen Bedeutungsmöglichkeiten – hier umgesetzt als gewichtete Kombination von Vektoren. Die konkrete Bedeutung kommt demnach nicht dem Zielwort, sondern vielmehr dem jeweils ermittelten Kontextvektor zu. Dies deckt sich mit der Annahme, dass Bedeutung nur im kontextualisierten Wort zu finden ist: bei isolierter Betrachtung bleibt die kontextuelle Aktivierung aus; die Bedeutung bleibt somit unbestimmt.[127]

Offen ist nun noch die Frage, welche Auswirkung der hier beschrieben Prozess einer lokal wirksamen Bedeutungskonstitution auf die Darstellung des semantischen Potentials von Wörtern hat. Um dies zu beantworten, müssen die unterschiedlichen Kontextualisierungen zueinander in Beziehung gesetzt werden, um dadurch ein differenzierteres Bild des Bedeutungspotentials zu erhalten, als es durch die Kookkurrenzvektoren gegeben ist, denn diese enthalten die Bedeutungsmöglichkeiten nur implizit. Tatsächlich ist es von den zuletzt dargestellten Plots mit mehreren Verwendungen eines gleichen Wortes nur noch ein kleiner Schritt hin zu den semantischen Profilen, wie sie in Abschnitt 5.2.3 beschrieben wurden – diese sind Gegenstand des nachfolgenden Abschnitts.

7.3 Semantische Profile

Auf Grundlage der bisherigen Experimente kann aus den Bedeutungsvarianten, die sich aus der Verwendung in verschiedenen Kontexten ergeben, ein erweitertes Bedeutungsprofil erstellt werden. Nach der gezielten Betrachtung von Einzelvorkommen steht hier somit die Frage im Mittelpunkt, wie sich die lokal erzeugten Repräsentationen zueinander verhalten. Im Folgenden wird hierfür ein entsprechendes Experiment beschrieben, bei dem die Kontextvektoren mittels einer mehrstufigen Clusteranalyse strukturiert werden. Aus dieser Ausdifferenzierung ergibt sich für jedes Wort ein semantisches Profil, das seine Bedeutungsmöglichkeiten widerspiegelt.

127 Eine Ausnahme bilden all jene Fälle, in denen keine Kontextelemente mit ausreichend hoher Signifikanz identifiziert werden; dort wird der Ausgangsvektor im Sinne einer »default interpretation« als eine Art Grundbedeutung gedeutet.

Analog zu den bisherigen Experimentbeschreibungen wird auch hier zunächst der Versuchsaufbau (Abschnitt 7.3.1) beschrieben sowie die konkrete Parametrisierung erläutert (Abschnitt 7.3.2). Im Anschluss daran werden die Ergebnisse verschiedener Beispielanalysen präsentiert (Abschnitt 7.3.3). Neben der Analyse von Einzelwörtern können dabei zu Vergleichszwecken auch zwei oder mehrere Wörter in einem gemeinsamen Plot zusammengefasst werden.

7.3.1 Aufbau des Experiments

Analog zu dem Vorgehen in den vergangenen Abschnitten werden zunächst Kookkurrenzvektoren erstellt. Darauf aufbauend werden die einzelnen Vorkommen eines Wortes in eine vektorielle Repräsentation überführt, indem der Vektor eines gegebenen Zielworts mit den Vektoren der Kontextelemente kombiniert wird. Diese Kontextvektoren werden in einem zusätzlichen Verarbeitungsschritt mittels Clusteranalyse zu Gruppen ähnlicher Elemente zusammengefasst und abschließend visualisiert. Abbildung 7.9 zeigt den entsprechenden Versuchsaufbau für die Erstellung von semantischen Profilen.

Um die Lesbarkeit der Ergebnisse der Clusteranalyse zu erhöhen, werden die ermittelten Cluster zunächst verkleinert, so dass nicht alle Vektoren im abschließenden Plot erscheinen. In der zu diesem Zweck implementierten ClusterFilter-Komponente wird für jedes Cluster nur eine festgelegte Anzahl von Elementen behalten, die über einen entsprechenden Parameter eingestellt werden kann. Die Filterung der Elemente orientiert sich dabei an den Clusterzentren, das heißt, es werden jeweils nur die Elemente mit der größten Ähnlichkeit zum jeweiligen Zentroid akzeptiert – diese dienen damit als Stellvertreter für eine Gruppe von ähnlichen Kontexten. Vor der Visualisierung werden die gefilterten Vektoren durch eine weitere Clusteranalyse erneut zueinander in Beziehung gesetzt, um dadurch auch die Beziehungen der ermittelten Cluster untereinander herauszuarbeiten.

7.3.2 Parametrisierung

Tabelle 7.3 zeigt die im Experiment eingesetzte Parametrisierung der beteiligten Komponenten in den einzelnen Verfahrensschritten. Der zentrale Verfahrensschritt für die Erstellung der semantischen Profile ist eine Clusteranalyse der Kontextvektoren. Wie in Abschnitt 6.2.7 beschrieben, gelten dichtebasierte Verfahren als besser geeignet für hochdimensionale Repräsentationen. Im Zuge des Experiments wird deshalb der DBSCAN-Algorithmus in der Implementation des ELKI-Frameworks genutzt. Im Unterschied zu distanzbasierten Verfahren muss die erwartete Clusterzahl in DBSCAN nicht vorgegeben werden. Im

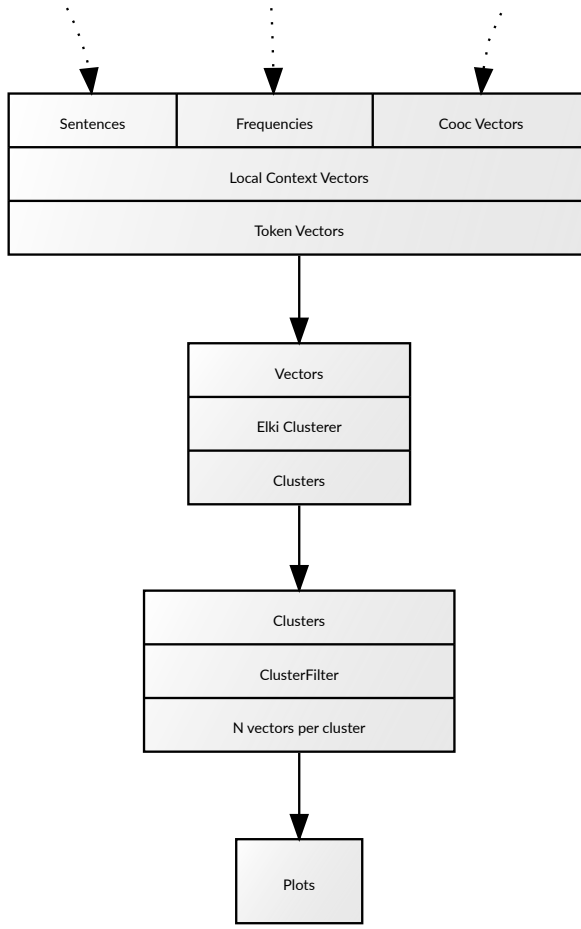


Abbildung 7.9: Schematischer Aufbau des Experiments zur Erstellung von semantischen Profilen in Tesla. Die von der Local Context Vectors-Komponente erzeugten Kontextvektoren werden mittels einer mehrstufigen Clusteranalyse strukturiert und anschließend über die integrierte Plotting-Funktion visualisiert.

Zusammenhang des Experiments ist dies ein großer Vorteil: Ziel ist es, die typischen Verwendungsweisen von Wörtern herauszuarbeiten, indem ähnliche Verwendungen zusammengefasst und damit von abweichenden Kontexten differenziert werden. Vor dem Hintergrund der Annahme, dass die Bedeutungskonstitution bei ambigen Wörtern zu einer stärkeren Differenzierung der Bedeutungsmöglichkeiten führt, lässt sich hier als Erwartungswert formulieren, dass sich bei mehrdeutigen Wörtern eine größere Zahl verschiedener

Semantische Profile		
Korpus	SdeWaC, 1 Mio. Sätze, deutsch	
	Stemming	ja
Vektoren	Fenster	3
	Länge	8.000 (häufigste)
	Anzahl	10.000 (ohne 100 häufigste)
	Gewichtung	PMI, normalisiert
CollocationVectors	Kontextelemente	1
	Assoziationsmaß	Log-Likelihood
	Positionsgewichtung	ja
	Normalisierung	Am max. Signifikanzwert
Clusteranalyse	ELKI DBSCAN	
	Max. Radius	0,25
	Min. Clustergröße	5
ClusterFilter	Anzahl Elemente	1
	Min. Clustergröße	1

Tabelle 7.3: Konfiguration der Komponenten im Experiment zur Erstellung von semantischen Profilen. Die Parameter zur Erstellung der Kookkurrenzvektoren sowie der Kontextvektoren wurden aus den vorangehenden Abschnitten übernommen.

Verwendungsweisen abgrenzen lässt, so dass die resultierenden semantischen Profile verschiedene Grade an Heterogenität aufweisen.

Im DBSCAN-Algorithmus wird die Clusterzahl über zwei Parameter beeinflusst: zum einen kann die maximal tolerierte Distanz der Datenpunkte im Vektorraum angegeben werden, so dass nur Elemente innerhalb des damit beschriebenen Radius als Kandidaten für die Bildung eines Clusters zugelassen werden. Zum anderen kann die Mindestgröße der Cluster festgelegt werden; nur wenn sich innerhalb des angegebenen Radius eine ausreichende Anzahl von Elementen findet, werden sie zu einem Cluster zusammengefasst.[128] In der hier gewählten Konfiguration werden verhältnismäßig kleine Cluster zugelassen, die oftmals nur

128 Dadurch dass der Abstand immer nur zwischen zwei Datenpunkten ermittelt wird, sind die Cluster – anders als bei rein distanzbasierten Verfahren wie etwa dem K-Means-Algorithmus – nicht zwangsläufig sphärisch bzw. kreisförmig organisiert. Hier zeigt sich der Vorteil dichtebasierter Verfahren bei hochdimensionalen Daten, insofern der Wortraum nicht symmetrisch organisiert ist.

aus gleichen oder besonders ähnlichen Kontexten bestehen. Bei der anschließenden Anwendung des ClusterFilter wird für jedes Cluster nur ein Element behalten, das stellvertretend für das jeweilige Cluster steht. Beim anschließenden hierarchischen Clustering, das im Zuge der Visualisierung durchgeführt wird, werden die verbliebenen Kontextvektoren nochmals zusammengefasst, um dadurch gegebenenfalls vorhandene typische Verwendungsweisen identifizieren zu können. Inwiefern dies möglich ist, sollen die Beispielanalysen zeigen, deren Ergebnisse im folgenden Abschnitt präsentiert werden.

7.3.3 Beispielanalysen

Im Folgenden werden verschiedene Beispielanalysen auf Grundlage des oben beschriebenen Workflows durchgeführt. Um eine bessere Vergleichbarkeit der Ergebnisse herzustellen, wird dabei eine Mengenbeschränkung auf maximal 1.000 Kontexte je Wort vorgenommen. Dadurch werden zum einen die Unterschiede in der Frequenz ausgeglichen, zum anderen sorgt dies für eine höhere Lesbarkeit der resultierenden Plots. Abbildung 7.10 zeigt zunächst das Ergebnis des Experiments für das Beispielwort *spielen*. Wie im vergangenen Abschnitt beschrieben, werden die Kontexte zunächst mittels Clusteranalyse und Filterung auf eine geringere Anzahl besonders typischer Verwendungen reduziert, die jeweils stellvertretend für weitere ähnliche Verwendungskontexte stehen.

Die Abbildung zeigt das Ergebnis des zweiten, hierarchischen Clusterings, das im Zuge der Visualisierung mit R durchgeführt wird. Die Verzweigungen im Dendrogramm (unten) spiegeln die Beziehungen der verbliebenen Elemente untereinander wider. Durch einen ›Schnitt‹ nahe der Wurzel (horizontale Linie am oberen Rand) ergibt sich ein flaches Clustering, das durch eine zusätzliche farbige Markierung auf den Scatterplot (oben) übertragen werden kann. Da hier – anders als bei der Projektion in einen Referenzraum – nur die Kontextvektoren selbst abgebildet werden, tritt die spezifische Verteilung der Kontexte innerhalb des Wortraums deutlicher hervor. Dadurch wird die interne Struktur des Verwendungsmusters sichtbar: Neben der Kollokation *Rolle spielen* sind hier vor allem die Verwendungen im Sinne der Themenbereiche »Fußball« und »Musik« deutlich erkennbar. Das spezifische Muster, das sich daraus ergibt, wird noch deutlicher, wenn das gleiche Ergebnis in eine andere Darstellung gebracht wird (Abb. 7.11).[129]

In dieser Darstellung bilden sich verschiedene Zweige heraus, auf denen Gruppen ähnlicher Verwendungen zusammengefasst sind. Da bei datenbasierten

129 In jedem Durchlauf wird für das Ergebnis eine Vielzahl solcher Darstellungsvarianten erstellt (u.a. verschiedene Dendrogramme, aber auch phylogenetische Bäume in verschiedenen Layouts, vgl. dazu Abschnitt 6.2.8), die für die hier gezeigten Experimente jedoch keine unmittelbare Rolle spielen. Sie sind vielmehr dem nebenläufigen Ziel dieser Arbeit verpflichtet, ein möglichst umfassendes Analyseinstrument für distributionell motivierte Untersuchungen zu erstellen.

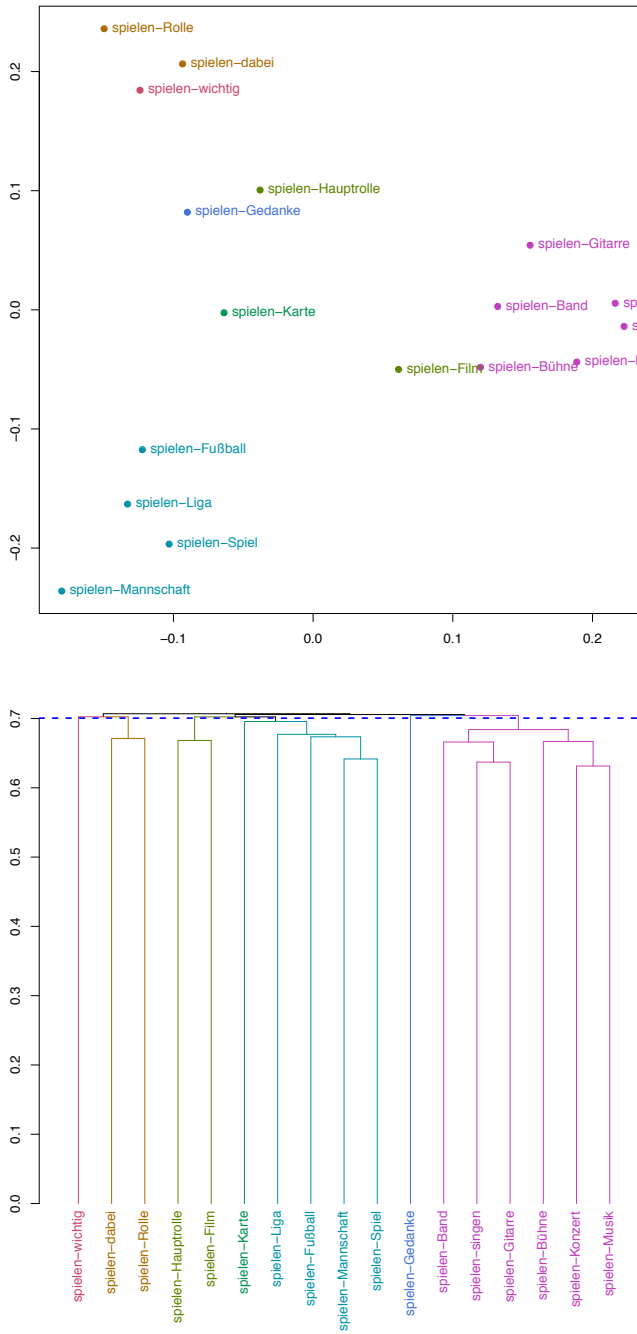


Abbildung 7.10: Typische Verwendungen für das Beispielwort *spielen*, dargestellt als Scatterplot sowie als Dendrogramm.

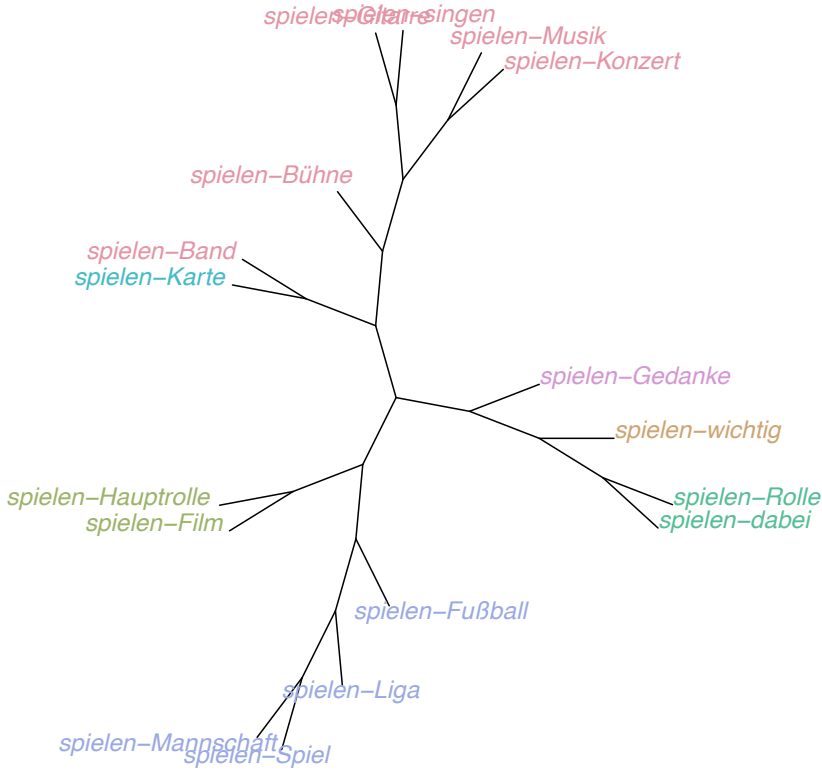


Abbildung 7.11: Typische Verwendungen für das Beispielwort *spielen*, hier dargestellt als »unrooted neighbor-joining tree«.

Verfahren wie dem hier beschriebenen die Abhängigkeit von der eingesetzten Datengrundlage sehr groß ist, ergibt sich bei der Verwendung eines anderen Korpus ein anderes Gesamtbild. Abbildung 7.12 zeigt das entsprechende Muster bei einer Analyse auf Grundlage des LCC-Korpus. Auch bei einem Wechsel der Datengrundlage bildet sich ein ähnliches Profil heraus. So lässt sich auch hier unter anderem ein Cluster für *Fußball* identifizieren, im Vergleich zu Abbildung 7.11 kommt jedoch zusätzlich noch ein eigenes Cluster für weitere Sportarten hinzu (unten rechts), was darauf zurückzuführen ist, dass im LCC-Korpus offenbar eine größere Zahl entsprechender Beispiele enthalten ist.

Ein solches Muster bildet sich für jedes Wort in anderer Weise heraus, so dass sich anhand der Muster die Unterschiede bezüglich der Verwendungsmöglichkeiten verschiedener Wörter illustrieren lassen. Abbildung 7.13 zeigt eine Gegenüberstellung der entsprechenden Muster für die Beispielwörter *scharf* und *Krone*. Zugunsten der Lesbarkeit wurde hierbei eine restriktivere Filterung angewendet. Die stärkere Filterung führt dazu, dass deutlich weniger Elemente im

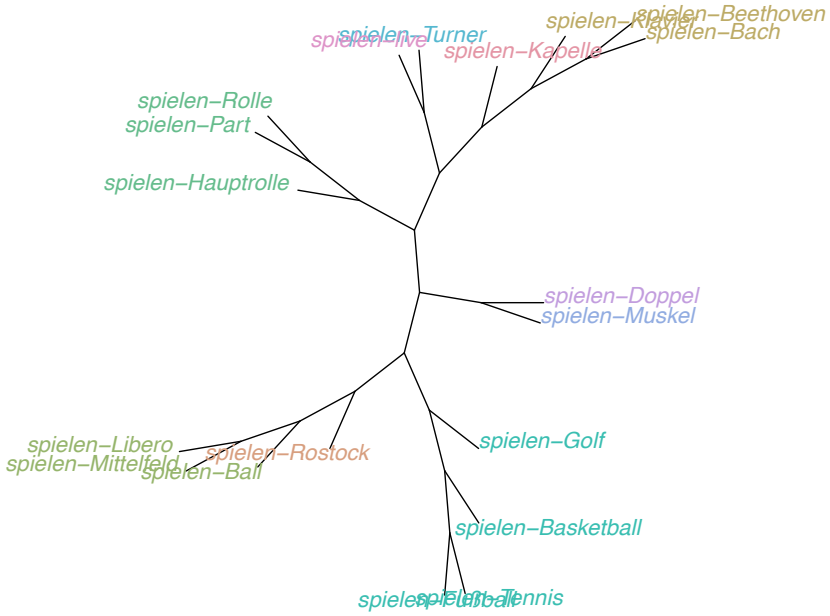


Abbildung 7.12: Semantisches Profil von *spielen* auf Grundlage des LCC-Korpus. Da das Korpus andere Beispiele enthält, ergibt sich ein abweichendes Profil; die Gemeinsamkeiten zu Abbildung 7.11 sind dennoch deutlich erkennbar.

Plot erscheinen.[130] Die verschiedenen Verwendungsweisen lassen sich dennoch deutlich unterscheiden – etwa *scharfe Kritik* gegenüber *scharfes Messer* im oberen Plot, oder die *Baumkrone* im Gegensatz zur *Währung* im unteren Plot. Die verbliebenen Elemente können damit als die besonders typischen Verwendungen des jeweiligen Wortes angesehen werden.

Jedes der Profile entspricht dabei einem bestimmten Bereich des Wortraums, der als eine erweiterte Repräsentation des jeweiligen Bedeutungspotentials verstanden werden kann. Wie die gemeinsame Projektion mehrerer Beispielwörter in Abbildung 7.14 zeigt, ist auch die Abgrenzung von anderen Potentialen nach wie vor gegeben.

Die interne Ausdifferenzierung der einzelnen Potentiale beruht dabei auf der Ähnlichkeit der Kontextwörter untereinander. Ausschlaggebend sind in der hier beschriebenen Umsetzung damit einzig die Kollokate: Die durch sie

130 Die Filterung wird verstärkt, indem hier der Parameter für die Mindestgröße der Cluster in der DBSCAN-Komponente auf 10 Elemente erhöht wird, wodurch insgesamt weniger Cluster gebildet werden. Nach Anwendung des ClusterFilter bleiben hier entsprechend weniger Stellvertreter-Elemente übrig. Umgekehrt würden bei geringerer Mindestgröße deutlich mehr Cluster gefunden, so dass eine feinere Differenzierung möglich wird – was jedoch zu unleserlichen Plots führt.

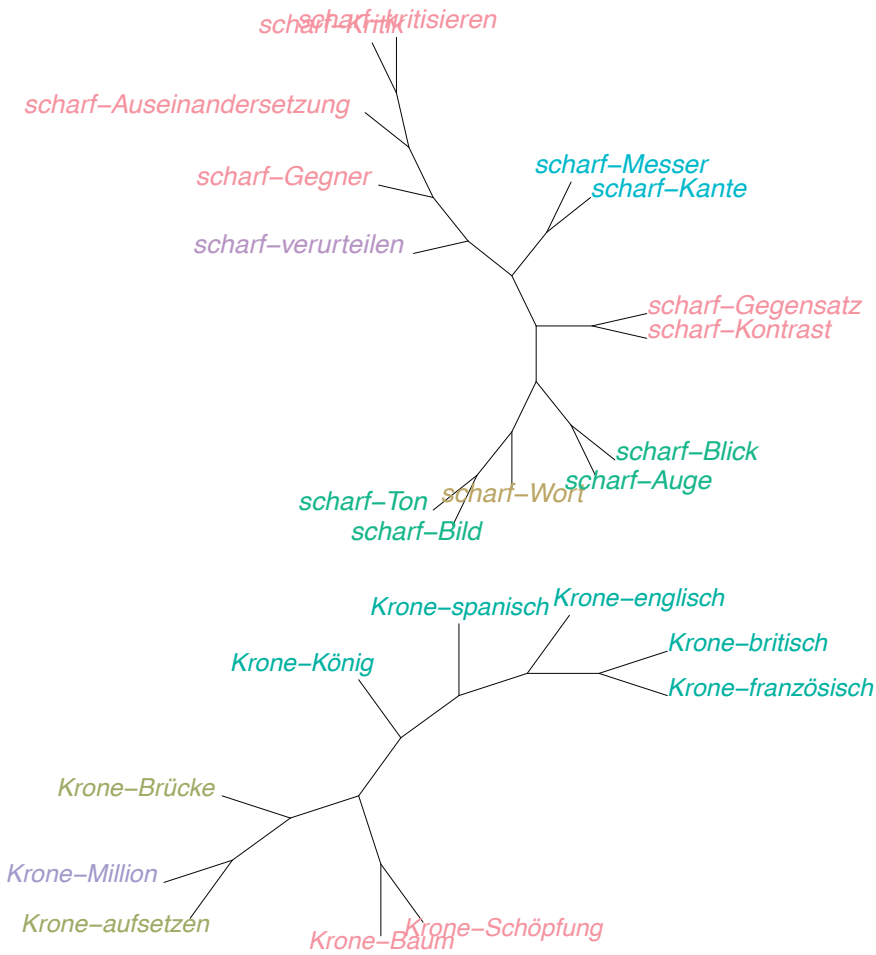


Abbildung 7.13: Jedes Wort bildet ein spezifisches Profil heraus, hier illustriert am Beispiel der Wörter *scharf* und *Krone*. Die geringere Anzahl an Elementen ergibt sich aus einer Erhöhung des Parameters für die Mindestgröße der Cluster im DBSCAN-Algorithmus.

ausgelöste Bedeutungskonstitution bestimmt letztlich die Verteilung der einzelnen Verwendungen innerhalb der Profile, da sie für verschiedene Kontexte jeweils anders ausfällt. Dadurch dass die Kontextvektoren auf einer gemeinsamen Ausgangsrepräsentation beruhen, bleiben sie dennoch in einer gemeinsamen Region des Wortraums organisiert. Die Ähnlichkeiten der einzelnen Bereiche untereinander entsprechen dabei im Wesentlichen den Ähnlichkeiten der zugrunde gelegten Kookkurrenzvektoren, so dass die Relationen zu anderen Wörtern (bzw. zu deren Bedeutungspotentialen) auch bei der hier vorgenommenen Ausdifferenzierung grundsätzlich erhalten bleiben.

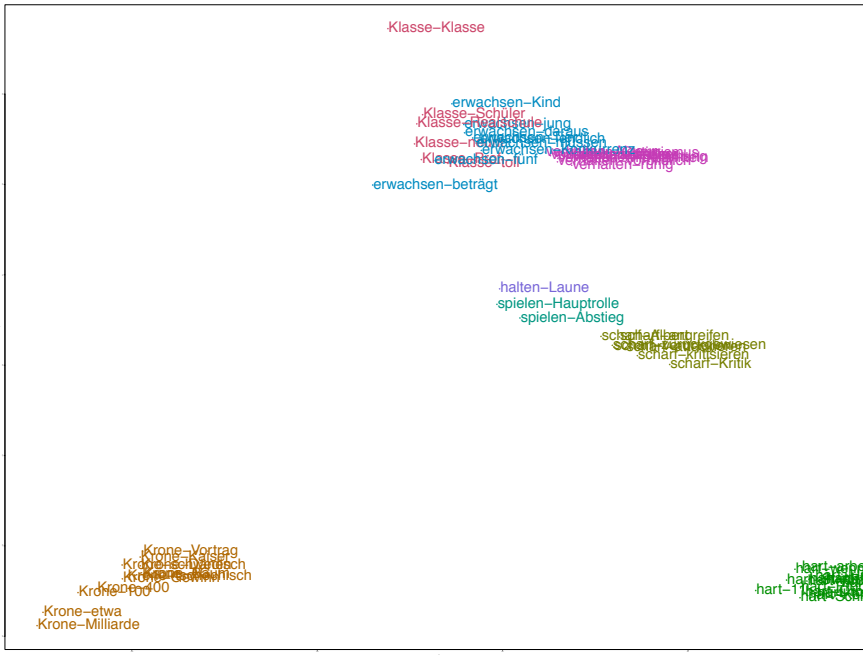


Abbildung 7.14: Verteilung semantischer Profile im Vektorraum (hier der Beispielwörter aus Abschnitt 6.2.9). Es bilden sich voneinander abgrenzbare Bereiche, die bei starker Ähnlichkeit auch überlappen können.

7.4 Zusammenfassung

In diesem Kapitel wurde eine Reihe von Experimenten beschrieben, in denen schrittweise die verschiedenen Aspekte des in dieser Arbeit entworfenen Modells der Bedeutungskonstitution umgesetzt werden. Ausgangspunkt aller Experimente ist die Erstellung von Kookkurrenzvektoren,[131] auf deren Grundlage kleine Ausschnitte des Gesamtdatenraums als Referenzräume für die Visualisierung der Bedeutungskonstitution erstellt werden (siehe Abschnitt 7.1). In zwei darauf aufbauenden Experimenten wurde anschließend die eigentliche Umsetzung des Modells

131 In den Experimenten werden die Vektoren mit einer Kontextbreite von drei Elementen, einer Beschränkung des Merkmalssets auf die 8.000 häufigsten Wörter sowie einer Gewichtung mittels »pointwise mutual information« erstellt. Die hier gewählte Parametrisierung beruht dabei auf einer Abwägung zwischen möglichst viel Information bei möglichst geringer Länge, um die Anzahl rechenintensiver Vergleichsoperationen auf ein geringes Maß zu reduzieren. Grundsätzlich besteht eine sehr große Vielfalt an Konfigurationsmöglichkeiten. Die Frage nach optimalen Parametern ist jedoch nicht Gegenstand dieser Arbeit, hierfür sei z.B. auf Turney/Pantel (2010) oder Levy/Bullinaria (2001) verwiesen.

der Bedeutungskonstitution beschrieben. Im Mittelpunkt des ersten Experiments steht der Prozess selbst: Ausgehend von der Annahme, dass die Kookkurrenzvektoren jeweils die gesamten Bedeutungsmöglichkeiten des durch sie repräsentierten Wortes implizieren, wurde die Bedeutungskonstitution hier im Sinne einer lokal wirksamen kontextuellen Aktivierung von Teilen des Bedeutungspotentials umgesetzt (siehe Abschnitt 7.2). Hierbei wird in jedem Kontext der Kookkurrenzvektor eines Zielworts mit den Vektoren der Kontextelemente kombiniert. Um zu berücksichtigen, dass gemäß dem Modell nicht alle Wörter den gleichen Einfluss auf den Prozess haben, wird zuvor noch eine Gewichtung der Kontexte nach Signifikanz vorgenommen. In der hier gewählten Konfiguration wurde der Kontext jeweils auf das signifikanteste Element beschränkt, so dass die einzelnen Kontextualisierungen hier durch mehr oder weniger starke Kollokationen repräsentiert werden.

Um die durch die Kombination mit den Kollokaten hervorgerufene Veränderung der Repräsentation sichtbar zu machen und damit den Prozess der Bedeutungskonstitution zu veranschaulichen, wurden ausgewählte Beispielkontexte in einen Referenzraum projiziert, der aus den ähnlichsten Wörtern des jeweils betrachteten Zielworts besteht.[132] Dadurch konnte offengelegt werden, dass mit der Veränderung der Repräsentation auch eine systematische Veränderung der Position im Referenzraum einhergeht, bei der die erzeugten Kontextvektoren in Richtung des jeweiligen Kollokats bewegt werden. In den Beispielanalysen zeigt sich dies immer dann besonders deutlich, wenn die betreffenden Kollokate im Referenzraum mit enthalten sind; daraus, dass im Gesamtdatenraum alle potentiellen Kollokate enthalten sind, kann jedoch geschlossen werden, dass eine entsprechende Bewegung in jedem Kontext stattfindet.

Darauf aufbauend wurden in einem zweiten Experiment die verschiedenen Kontextualisierungen strukturiert und zu einem semantischen Profil zusammengefasst (siehe Abschnitt 7.3). Im Zuge des Experiments werden die Repräsentationen der Kontexte durch Anwendung einer flachen Clusteranalyse sowie einer anschließenden Filterung zunächst auf eine geringere Anzahl reduziert. Übrig bleiben typische Verwendungen, die jeweils durch einen Stellvertreter repräsentiert sind. In einem zweiten, diesmal hierarchischen Clustering werden die verbliebenen Elemente zueinander in Beziehung gesetzt. Daraus ergibt sich für jedes Wort ein spezifisches Muster, das seine Bedeutungsmöglichkeiten widerspiegelt und in diesem Sinne als eine Ausdifferenzierung des Bedeutungspotentials angesehen werden kann. Entsprechend der Vorhersage des Modells belegen die Vektoren der verschiedenen Kontextualisierungen eine weitgehend zusammenhängende Region im Vektorraum. Diese lässt sich von anderen Regionen abgrenzen,

132 Neben der in dieser Arbeit verfolgten Variante gibt es weitere Möglichkeiten für die Erstellung eines Referenzraums: Anstelle der ähnlichsten Elemente könnten die Referenzräume beispielsweise auch durch die signifikantesten Kollokate definiert werden, oder es könnte eine feste Anzahl von Elementen aus einer vorherigen Clusteranalyse als Grundlage für einen gemeinsamen Referenzraum dienen.

wobei es bei ausreichend hoher Ähnlichkeit der Kollokationen auch zu Überlappungen kommen kann.

Zusammenfassend lässt sich hier festhalten, dass sich in den Experimenten die durch das Modell vorhergesagten Effekte weitgehend bestätigen: Zum einen kann die Bedeutungsvariation durch eine kontextuell bedingte Veränderung der Repräsentation sichtbar gemacht werden, zum anderen zeigen sowohl die Projektionen in den Referenzraum als auch die erstellten semantischen Profile, dass die jeweiligen Veränderungen einem konsistenten Muster folgen. Welche Rückschlüsse das in Bezug auf die theoretischen Annahmen zulässt, wird im abschließenden Kapitel diskutiert.

8. Fazit: Muster und Bedeutung

Das wesentliche Ziel dieser Arbeit bestand in einer computerlinguistischen Modellierung des Prozesses der Bedeutungskonstitution sowie der anschließenden softwaretechnologischen Umsetzung und experimentellen Überprüfung des Modells. Auf Grundlage des *dynamic construal approach* (Croft/Cruse 2004) wird die Bedeutungskonstitution in dieser Arbeit als dynamischer Prozess verstanden, bei dem sich die Bedeutung sprachlicher Einheiten erst innerhalb lokaler Kontexte in Relation zu deren allgemeinem Bedeutungspotential konkretisiert. Die Modellierung stützt sich auf das Word Space Model (WSM) nach Schütze (1992; 1993), dessen Vektoren hier als Repräsentation der Bedeutungspotentiale ausgelegt wurden. Der Prozess wurde darauf aufbauend in Anlehnung an Marr (1982) als informationsverarbeitender Prozess modelliert, im Zuge dessen eine Transformation der Ausgangsrepräsentation durch Hinzunahme der Vektoren der Kontextelemente erfolgt. Die softwaretechnologische Umsetzung des Modells erfolgte schließlich auf Grundlage des Text Engineering Software Laboratory (Tesla), das in dieser Arbeit die Funktion eines virtuellen Labors übernimmt, in dem das Modell in einer Reihe von Experimenten erprobt werden konnte, um Rückschlüsse auf den explikativen Wert der zugrunde gelegten Konzeption eines dynamischen Bedeutungsbegriffs zu ziehen. Vor dem Hintergrund der Ergebnisse in Kapitel 7 wird im Folgenden nochmals eine abschließende Bewertung des methodischen Vorgehens vorgenommen.

In den Experimenten konnte anhand von Beispielanalysen ausgewählter Wörter gezeigt werden, dass die Modellierung des Prozesses der Bedeutungskonstitution als kontextuelle Aktivierung im Vektorraum es ermöglicht, das Phänomen der Bedeutungsvariation erfolgreich zu simulieren. So kann in den Beispielanalysen unter anderem nachvollzogen werden, dass bei einer mehrfachen Repräsentation von Wörtern auf Grundlage ihrer Kontexte (hier eingeschränkt auf Kollokationen) verschiedene Bedeutungen abgeleitet werden können. Eine Ausnahme stellen all jene Fälle dar, in denen es nicht möglich ist, Kontextelemente mit einer ausreichend hohen Signifikanz in Bezug auf das gemeinsame Auftreten zu identifizieren; dort kann jedoch der Ausgangsvektor als eine Art Grundbedeutung interpretiert werden. Vor allem die Tatsache, dass die Veränderung systematisch passiert, gibt deutliche Hinweise darauf, dass die Annahme einer Bedeutungskonstitution als zentrales Element eines dynamischen Bedeutungsbegriffs eine konsistente Erklärung der in Sprache beobachtbaren Bedeutungsvariation ermöglicht.

Die Ergebnisse der Experimente sind damit konform zu der Annahme der Kognitiven Semantik, dass Wörter für sich genommen nur über ein unausgedeutetes Bedeutungspotential verfügen (im Modell durch einen einfachen Kookkurrenzvektor repräsentiert) und dass die konkrete Bedeutung erst im Zuge einer Bedeutungskonstitution temporär zugewiesen wird (im Modell durch die Kombination mit den Vektoren der Kollokate). Eine Differenzierung der Bedeutung ist

demnach nur für das kontextualisierte Wort möglich, was in den Experimenten am Beispiel von Kollokationen gezeigt werden konnte. Das einzelne Wort ist in dieser Sicht nur als ein Baustein für das Herausbilden der konkreten Bedeutung anzusehen. Vergleichbar zu Buchstaben ermöglichen sie einen weitgehend flexiblen Einsatz, der jedoch durch ihre jeweils spezifischen Kombinationsmöglichkeiten eingeschränkt wird. In dieser Perspektive sind Bedeutungen nicht als solche im mentalen Lexikon hinterlegt und werden demnach nicht einfach bei Bedarf ›abgerufen‹, sondern sie bilden sich in Abhängigkeit verschiedener Kontextualisierungen jedes Mal neu und jeweils unterschiedlich heraus.

Wenngleich die Experimente somit die Annahme einer Bedeutungskonstitution als Grundlage für einen dynamischen Bedeutungsbegriff unterstützen, so ist dies in Bezug auf die Kognitive Linguistik insgesamt nicht ohne weiteres möglich. Dies liegt in erster Linie in der methodisch bedingten Beschränkung auf Sprachdaten begründet. Ein wichtiger Aspekt, der bei einer Modellierung auf Grundlage von Korpora nur bedingt berücksichtigt werden kann, ist die zentrale Rolle des Sprechers in der Konzeption der Kognitiven Linguistik. Vor dem Hintergrund der These des »Embodiment«, derzufolge unsere kognitiven Fähigkeiten unmittelbar mit den physischen Bedingungen unserer körperlichen Existenz in der Welt zusammenhängen und somit auch einen entsprechenden Einfluss auf Verstehensprozesse haben, müssten aus Sicht der Kognitiven Linguistik hier auch außersprachliche Faktoren in die Modellierung einbezogen werden – was im Zusammenhang mit der in dieser Arbeit eingesetzten distributionellen Methodik nicht umsetzbar wäre. Mit der Möglichkeit einer individuellen, situationsabhängigen Interpretation durch den Sprecher lässt die Kognitive Linguistik hier in gewissem Sinne einen im Modell nicht erklärbaren Rest offen, der bei einer entsprechend strengen Auslegung mit einem rein distributionellen Vorgehen nicht abgedeckt werden kann.

Akzeptiert man jedoch die Einschränkung auf textuelle Daten, so kann der hier modellierte Prozess der Bedeutungskonstitution zumindest im Ansatz als ein (wenngleich einfaches) Modell der Interpretation durch den Sprecher angesehen werden, die im Zuge der Kontextualisierung erfolgt. Von zentraler Bedeutung für das Modell sind dabei die Verwendungsmuster, die durch die Vektoren des Wortraums kodiert sind. Diese wurden im Rahmen der Arbeit umgedeutet: Statt als Repräsentation von vollwertigen Bedeutungen werden sie hier nur als vorläufige Strukturen angesehen, die das Bedeutungspotential von Wörtern repräsentieren. Die Verwendungsmuster können so als ein Bestandteil des semantischen Gedächtnisses verstanden werden, auf Grundlage dessen die konkreten Bedeutungen lokal gebildet werden.

Mit der Umdeutung der Kookkurrenzvektoren und der Erweiterung des WSM um den Prozess der Bedeutungskonstitution wurde im Rahmen dieser Arbeit eine Auslegung des Wortraums vorgenommen, die von der üblichen, klassisch strukturalistisch geprägten Deutung, wie sie zum Beispiel von Sahlgren (2006; 2008) vertreten wird, abweicht. Das WSM als solches wurde dabei nicht verändert,

vielmehr wurden die durch Vektoren repräsentierten Verwendungsmuster in den theoretischen Kontext der Kognitiven Semantik übertragen. Zwar sind die Verwendungsmuster allein nicht geeignet, um den dynamischen Bedeutungsbegriff der Kognitiven Semantik zu erklären, da Bedeutungen aus Sicht der Kognitiven Semantik nicht unabhängig vom Kontext existieren und deshalb nicht unmittelbar repräsentiert werden können; dennoch spielen die Verwendungsmuster von Wörtern auch in kognitiver Perspektive eine entscheidende Rolle, indem sie den Ausgangspunkt für den Prozess der Bedeutungskonstitution bilden. In der hier vorgeschlagenen Modellierung sind die Verwendungsmuster der entscheidende Informationsträger und -lieferant. Mit anderen Worten: ohne Muster keine Bedeutung.

Durch die Umsetzung in Tesla konnte zudem ein weiteres Ziel der Arbeit eingelöst werden, das in der Bereitstellung von Komponenten und Verfahren für distributionell motivierte Untersuchungen auch über diese Arbeit hinaus bestand. Zum einen lassen sich dadurch die bestehenden Experimente ausbauen, etwa um das Modell zu erweitern, indem verschiedene Ausnahmen berücksichtigt werden. Ungelöst ist hier zum Beispiel das Problem, dass die Kontextwörter selbst mehrdeutig sein können, so dass auch die Hinzunahme von Kollokaten nicht immer zu einer klaren Konkretisierung der Bedeutung führt.[133] Hier könnte ein Ansatz in der Hinzunahme weiterer Informationen bestehen, etwa indem die auf Basis von Kollokationen erstellten Kontextvektoren zusätzlich mit einem (geringer gewichteten) Vektor kombiniert werden, der auf Basis des gesamten Kontextes erstellt wird. Zum anderen können die Komponenten, in denen die einzelnen Verfahrensschritte umgesetzt wurden, auch für andere, weiterführende Experimente eingesetzt werden. So ist beispielsweise denkbar, aufbauend auf den Ergebnissen dieser Arbeit eine Klassifikation im Sinne einer Wortsinndisambiguierung zu entwerfen. Ausgangspunkt könnte beispielsweise das hier beschriebene Vorgehen zur Ermittlung semantischer Profile sein: aus deren interner Struktur lassen sich im Ansatz die verschiedenen Lesarten von Wörtern ableiten, welche extrahiert und als Basis für eine Disambiguierung eingesetzt werden könnten, vergleichbar etwa zu dem in Schütze (1998) beschriebenen Vorgehen.

Die im Rahmen dieser Arbeit implementierten Komponenten und Verfahren können somit als eine Art methodischer Werkzeugkasten für distributionell motivierte Untersuchungen angesehen werden. Gerade für die Kognitive Linguistik ist die distributionelle Methodik als eine vielversprechende Ergänzung anzusehen, da sie im Hinblick auf den durch sie propagierten gebrauchsorientierten Ansatz auf geeignete Analysetechniken angewiesen ist. In diesem Zusammenhang stellen linguistische Komponentensysteme wie Tesla eine wertvolle Unterstützung dar: zum einen bieten diese kontrollierte Bedingungen für Experimente sowie eine umfassende Dokumentation und gewährleisten damit die Reproduzierbarkeit von

133 So kann etwa die in den Beispielanalysen häufig auftretende Kombination von *Rolle* und *spielen* sowohl ›wichtig‹ bedeuten als auch ›schauspielern‹.

Experimenten; zum anderen können sie nach Bedarf mit spezialisierten, auf den jeweiligen Anwendungsfall zugeschnittenen Komponenten ausgestattet werden – so wie es auch im Rahmen dieser Arbeit für Tesla durchgeführt wurde.

In Rückbezug auf den ersten Satz der vorliegenden Arbeit lässt sich damit nochmals die Rolle der Computerlinguistik für die Kognitionswissenschaft verdeutlichen, auch und gerade in Bezug auf die Modellierung einer kognitiv motivierten Bedeutungstheorie: Indem sie die Simulation von kognitiven Prozessen ermöglicht und Werkzeuge für die empirisch-experimentelle Erprobung der zugehörigen Modelle anbietet, spielt die Computerlinguistik selbst eine zentrale Rolle in der linguistischen Theoriebildung. In diesem Sinne versteht sich auch die im Zuge dieser Arbeit vorgenommene computerlinguistische Modellierung der Bedeutungskonstitution als Beitrag auf dem Weg zu einem erweiterten Verständnis der semantischen Dynamik von Sprache.

A. Komponenten

In Ergänzung zu Kapitel 6 werden im Folgenden die Komponenten gelistet, die im Zuge dieser Arbeit entwickelt und eingesetzt wurden. Die Gliederung orientiert sich dabei an den verschiedenen Verarbeitungsphasen (siehe auch Abschnitt 6.2):

1. Daten einlesen (Abschnitt A.1)
2. Vorverarbeitung (Abschnitt A.2)
3. Erstellung von Kookkurrenzvektoren (Abschnitt A.3)
4. Normalisierung und Gewichtung (Abschnitt A.4)
5. Repräsentation von Einzelvorkommen (Abschnitt A.5)
6. Clusteranalyse (Abschnitt A.6)
7. Visualisierung (Abschnitt A.7)

Sofern nicht anders angegeben, wurden die Komponenten im Rahmen der Arbeit entwickelt und werden mit der Standard-Installation von Tesla verbreitet.

A.1 Reader

Die Reader-Komponenten sind für das Einlesen der Daten sowie deren Bereitstellung für die Verarbeitung zuständig. Das zugrunde gelegte Konzept einer Trennung von Inhalt und Auszeichnung wird in Schwiebert (2012, 91f. sowie 116f.) beschrieben.

A.1.1 LCC Reader

LCC Reader	
Konsumiert	Signale (Text)
Produziert	Sentence Detector, Tokenizer

A.1.2 SdeWaC Reader

SdeWaC Reader	
Konsumiert	Signale (Text)
Produziert	Sentence Detector, Tokenizer, Stemmer, POS Tagger

A.2 Vorverarbeitung

Da das SdeWaC-Korpus bereits vorverarbeitet vorliegt (siehe Abschnitt 6.2.1), ist eine Vorverarbeitung nur für das LCC-Korpus nötig. Für die Erkennung von Wortgrenzen wird hier der Simple Tokenizer eingesetzt. Auf die Satzgrenzenerkennung kann verzichtet werden, da das LCC-Korpus bereits in Sätze eingeteilt ist, so dass die ursprünglichen Satz-IDs genutzt werden können. Die produzierten Tokens dienen als Input für weitere Verarbeitungsschritte, in den Experimenten in dieser Arbeit insbesondere der Vektorerstellung.

A.2.1 Simple Tokenizer

Wie in Schwiebert (2012) beschrieben nutzt der Simple Tokenizer die durch den java.text.BreakIterator bereitgestellten Möglichkeiten zur Segmentierung von Texten in Sätze und Wörter auf Basis vorgegebener Spracheinstellungen.

Simple Tokenizer		
Konsumiert	Signale (Text)	
Produziert	Sentence Detector, Tokenizer	
Konfiguration	Locale	Zu verwendende Spracheinstellung
	Tag Whitespaces	Definiert, ob Leerzeichen annotiert werden sollen
	Ignore case on type id	Definiert, ob die Type-Id der Annotationen Groß- und Kleinschreibung unterscheiden soll
Autor	Stephan Schwiebert	

A.2.2 Tree Tagger Wrapper

Über den TreeTaggerWrapper steht der probabilistische, auf Basis von Entscheidungsbäumen operierende Part-Of-Speech-Tagger von Schmid (1994) in Tesla zur Verfügung (siehe dazu Schwiebert 2012, 263f.).

Tree Tagger Wrapper	
Konsumiert	Sentence Detector, Tokenizer
Produziert	Tesla POS Tagger

Konfiguration	Tree Tagger binary directory	Programmverzeichnis
	Tree Tagger model file	Das zu verwendende Sprachmodell
Lizenz	Frei für nicht-kommerzielle Anwendung	
Autor	Stephan Schwiebert	

A.2.3 Snowball Stemmer Wrapper

Als performantere Alternative zum Stemmer des TreeTagger bindet diese Komponente den SnowballStemmer[134] in Tesla ein.

Snowball Stemmer Wrapper		
Konsumiert	Tokenizer	
Produziert	Stemmer	
Konfiguration	Language	Das zu verwendende Sprachmodell

A.3 Vektorerstellung

Die VectorGenerator-Komponente ist für die Erstellung der Kookkurenzvektoren zuständig. Die Komponente erfordert neben einem tokenisierten Korpus optional verschiedene Filter zur Manipulation des Merkmalsets. So können beispielsweise mit dem FrequencyFilter nur Attribute mit einer vorgegebenen Mindestfrequenz zugelassen werden, bzw. nur die *n* häufigsten Wörter als Merkmale festgelegt werden. Zudem kann ein Attributset vorgegeben werden, das in einem separaten Verarbeitungsschritt vorab erstellt wurde.

A.3.1 Sentence Based Vector Generator

Der SentenceBasedVectorGenerator basiert auf dem bereits in Tesla vorhandenen WordVectorGenerator. Während letzterer vektorielle Repräsentationen für beliebige Annotationen auf Basis ihres Kontextes erstellen kann, ist der SentenceBasedVectorGenerator speziell auf die Verarbeitung der hier verwendeten Korpora ausgelegt, die aus zufällig angeordneten Einzelsätzen bestehen.

134 Siehe http://snowball.tartarus.org (Zugriff vom 21.02.2018).

Sentence Based Vector Generator		
Konsumiert	Sequence Annotator, Filter (0–n), FeatureSet (optional)	
Produziert	Labeled Vectors	
Konfiguration	Window Size	Größe des Kontextfensters, das über das zugrundeliegende Signal verschoben wird
	HAL weighting	Nachbarschaftsgewichtung
	Filters to match for vector entry	Die Anzahl der Filter, die eine Annotation akzeptieren müssen, damit sie im Kontext einer anderen Annotation berücksichtigt wird
	Filters to match for vector generation	Die Anzahl der Filter, die eine Annotation akzeptieren müssen, damit ein Vektor für sie erzeugt wird
Autoren	Sonja Subicin, Stephan Schwiebert, Claes Neuefeind	

Die Merkmalsauswahl erfolgt über Filter, die die Vektorerstellung beeinflussen. Dabei wird zwischen ContextFilter und CreationFilter unterschieden: Erstere legen fest, für welche Elemente Vektoren erstellt werden, Letztere dienen dagegen der Beschränkung des Merkmalssets, indem sie festlegen, für welche Kontextelemente die Kookkurrenz gezählt wird.

A.3.2 Punctuation Filter

Diese Komponente ermöglicht die gezielte Filterung der verschiedenen Token-Kategorien, die vom SimpleTokenizer produziert werden.

Punctuation Filter		
Konsumiert	Tokenizer	
Produziert	Type Filter	
Konfiguration	Filter punctuation	Akzeptiert nur die vom Tokenizer als »Word« oder »Numerical« annotierten Elemente
	Filter numericals	Filtert die vom Tokenizer erkannten Zahlen

A.3.3 Frequency Range Filter

Diese Komponente erzeugt einen Filter auf Basis der Häufigkeit oder des Rangs. Zusätzliche Funktion ist die Angabe von Mindest- und Höchstfrequenz sowie der

Anzahl der zu akzeptierenden Elemente. Dieser Filter wird im Rahmen der Arbeit dafür verwendet, nur die häufigsten Wörter als Teil des Merkmalssets zu akzeptieren bzw. nur für diese Vektoren zu erstellen.

Frequency Range Filter		
Konsumiert	Annotation Statistics	
Produziert	Type Filter	
Konfiguration	Threshold	Gibt die Mindestanzahl bzw. den geringsten (bzw. höchsten) Rang der Annotationen an, die gefiltert werden sollen
	Range	Anzahl der Annotationen bzw. Ränge, die ausgehend vom Threshold gefiltert werden
	Filter most frequent annotations	Definiert, ob die häufigsten oder seltensten Annotationen berücksichtigt werden sollen
	Invert matching	Definiert, ob die Matching-Strategie invertiert werden soll, so dass bspw. nicht die 100 häufigsten Annotationen, sondern alle anderen vom Filter akzeptiert werden
	Use Rank	Gibt an, ob statt der absoluten Häufigkeit der Rang der Annotationen verwendet werden soll

A.3.4 POSFilter

Über den POSFilter können, sofern vorhanden, die Annotationen anhand ihrer Parts-Of-Speech gefiltert werden. Dadurch können die Merkmale beispielsweise auf Nomen beschränkt werden. Grundlage ist das Stuttgart-Tübingen-Tagset (STTS)[135], das sowohl im hier eingesetzten SdeWaC-Korpus als auch – bei Verwendung des LCC-Korpus – vom TreeTaggerWrapper verwendet wird.

POSFilter		
Konsumiert	Anchored Element Generator	
Produziert	Type Filter	
Konfiguration	POS tags	Gibt die POS-Tags an, die gefiltert werden sollen
	Invert matching	Definiert, ob die Matching-Strategie invertiert werden soll, so dass bspw. alle Annotationen außer Verben akzeptiert werden

135 Siehe http://www.ims.uni-stuttgart.de/forschung/ressourcen/lexika/TagSets/stts-table.html (Zugriff vom 21.02.2018).

A.3.5 Wordlist Filter

Mit dem Wordlist Filter kann eine extern definierte Wortliste eingelesen werden. Die dort zeilenweise enthaltenen Wörter werden zunächst anhand der vorhandenen Tokens mit einer Type-Id versehen, auf deren Grundlage der Filter erzeugt wird.

Wordlist Filter		
Konsumiert	Anchored Element Generator	
Produziert	Type Filter	
Konfiguration	Filename	Gibt den Pfad zur Wortlisten-Datei an, die zeilenweise die Wörter enthält
	Comment symbol	Definiert anhand des angegebenen Zeichens, ob eine Zeile als Kommentar angesehen und deshalb ignoriert werden soll

A.4 Normalisierung und Gewichtung

A.4.1 VectorNormalization

Da die Kookkurrenzvektoren aufgrund der unterschiedlichen Frequenzen der Wörter geometrisch betrachtet stark voneinander abweichende Längen aufweisen können, ist eine Längennormalisierung nötig. In der VectorNormalization-Komponente erfolgt die Normalisierung auf Grundlage der jeweiligen euklidischen Länge (siehe Abschnitt 6.2.4). Die Komponente enthält – wie auch alle anderen der nachfolgenden Vektorkomponenten – die Möglichkeit zur Visualisierung der erzeugten Vektoren.

VectorNormalization		
Konsumiert	Vector Generator, FeatureSet	
Produziert	Labeled Vectors	
Konfiguration	Plot type	Gibt die Art der optionalen Visualisierung an
	No of Elements	Definiert, wie viele Elemente im Plot erscheinen sollen

A.4.2 VectorWeighting

Diese Komponente stellt verschiedene Formen der Gewichtung bereit (siehe dazu Abschnitt 6.2.5). Die Vektoren können zudem direkt in der Komponente auch

normalisiert werden. Analog zur VectorNormalization kann zudem eine Visualisierung gewählt werden.

VectorWeighting		
Konsumiert	Vector Generator, FeatureSet	
Produziert	Labeled Vectors	
Konfiguration	Weighting scheme	Gibt das zu verwendende Gewichtungsmaß an, das auf die Vektoren angewendet werden soll
	Euclidean length	Optionale Möglichkeit zur Normalisierung
	Plot type	Gibt die Art der optionalen Visualisierung an
	No of Elements	Definiert, wie viele Elemente im Plot erscheinen sollen

A.5 Repräsentation von Einzelvorkommen

Die Vektorerstellung auf Grundlage lokaler Kontexte ist der zentrale Verfahrensschritt in dieser Arbeit. Für die Repräsentation von Einzelvorkommen wurden drei Komponenten implementiert, die sich in der Art und Weise unterscheiden, wie der Kontext definiert wird. Das Vorgehen orientiert sich im Wesentlichen am Konzept der Kookkurrenz zweiter Ordnung (siehe dazu Abschnitt 4.1.3).

A.5.1 CollocationVectors

Durch die CollocationVectors-Komponente wird das in dieser Arbeit vorgeschlagene Modell der Bedeutungskonstitution umgesetzt. In jedem Kontext wird zunächst durch die Anwendung von Assoziationsmaßen (siehe Abschnitt 6.2.6 sowie Anhang C) eine Gewichtung anhand der Signifikanz des gemeinsamen Auftretens vorgenommen, die als Grundlage für die Definition des Kontextes dient. Anschließend werden die Vektoren der signifikantesten Elemente mit dem Zielwortvektor kombiniert.

Mit Hilfe der CoocHelper-Komponente wird separat eine Indexstruktur der Kookkurrenzen in den einzelnen Kontexten erstellt, die für die Berechnung der Signifikanz benötigt werden, damit sie bei einer veränderten Parametrisierung nicht jedes Mal neu berechnet werden müssen. Konzeptionell entspricht das Vorgehen der Erstellung eines einzelnen Kookkurrenzvektors ohne vorherige Merkmalsauswahl.

Collocation Vectors		
Konsumiert	Tokenizer/Stemmer, Sentence Detector, Annotation Statistics, Vectors, FeatureSet, CoocsAndPositions	
Produziert	Labeled Vectors	
Konfiguration	Target word(s)	Ein oder mehrere Zielwörter, deren Kontexte verarbeitet werden sollen
	No. of context elements	Anzahl der signifikantesten Kontextelemente, die berücksichtigt werden sollen
	Position Weighting	Gewichtung in Abhängigkeit der Entfernung zum Zielwort, logarithmisch geglättet
	Sig. method	Assoziationsmaß für die Berechnung der Signifikanz
	Threshold	Mindestwert für Signifikanz
	Merge weight factor	Ermöglicht eine Verstärkung bzw. Abschwächung der errechneten Signifikanz
	Word list	Pfad zu externer Datei mit Filterkriterien für die zu verarbeitenden Kontexte
	Plot type	Art der optionalen Visualisierung
	No. of Elements	Anzahl der Elemente im Plot

CoocHelper		
Konsumiert	Tokenizer/Stemmer, Sentence Detector, Vectors, FeatureSet	
Produziert	CoocsAndPositions	
Konfiguration	Target word(s)	Ein oder mehrere Zielwörter, für die ein Index der Kookkurrenzen erstellt werden soll
	Max no. of contexts	Begrenzt die Anzahl der Kontexte, z.B. um eine einheitliche Grundlage für Vergleiche zu schaffen

A.5.2 Context Vectors

In der ContextVectors-Komponente werden in Anlehnung an Schütze (1998) für jedes Vorkommen eines angegebenen Wortes lokale Repräsentationen erzeugt, indem der Vektor des Zielworts mit dem Zentroid der Vektoren der Kontextelemente kombiniert wird. Anders als bei Schütze wird hierbei ein parametrisierbares Kontextfenster eingesetzt.

Context Vectors		
Konsumiert	Tokenizer/Stemmer, Sentence Detector, Annotation Statistics, Vectors, FeatureSet	
Produziert	Labeled Vectors	
Konfiguration	Target word(s)	Ein oder mehrere Zielwörter, deren Kontexte verarbeitet werden sollen
	Window size	Definiert die Kontextbreite
	Position Weighting	Gewichtung in Abhängigkeit der Entfernung zum Zielwort, logarithmisch geglättet
	Merge weight factor	Verstärkung bzw. Abschwächung des Zentroids bei der Zusammenführung mit dem Zielwortvektor
	Word list	Pfad zu externer Datei mit Filterkriterien für die zu verarbeitenden Kontexte
	Plot type	Art der optionalen Visualisierung
	No of Elements	Anzahl der Elemente im Plot

A.5.3 Sentence Vectors

Als vereinfachte Variante wurde zudem auch eine SentenceVectors-Komponente implementiert, bei der stets der ganze Satz als Kontext genommen wird. Die Satzvektoren sind damit gewissermaßen ein Spezialfall der Kontextvektoren.

Sentence Vectors		
Konsumiert	Tokenizer/Stemmer, Sentence Detector, Annotation Statistics, Vectors, FeatureSet	
Produziert	Labeled Vectors	
Konfiguration	Target word(s)	Ein oder mehrere Zielwörter, deren Kontexte verarbeitet werden sollen
	Position Weighting	Gewichtung in Abhängigkeit der Entfernung zum Zielwort, logarithmisch geglättet
	Merge weight factor	Verstärkung bzw. Abschwächung des Zentroids bei der Zusammenführung mit dem Zielwortvektor
	Word list	Pfad zu externer Datei mit Filterkriterien für die zu verarbeitenden Kontexte
	Plot type	Art der optionalen Visualisierung
	No of Elements	Anzahl der Elemente im Plot

A.6 Clustering

Als Erweiterung zu in Tesla bereits vorhandenen Verfahren für die Clusteranalyse (siehe Schwiebert, 2012), wurde im Zuge dieser Arbeit eine Reihe zusätzlicher flacher Cluster-Algorithmen eingebunden. Über das ELKI Data-Mining-Framework[136] wurden insgesamt acht verschiedene Verfahren eingebunden, wobei im Rahmen dieser Arbeit nur das dichtebasierte DBSCAN-Verfahren tatsächlich eingesetzt wird.

Um die verschiedenen Parametrisierungen angemessen abbilden zu können, wurden die adaptierten Verfahren als individuelle Komponenten implementiert. Das Clusterergebnis wird in Form eines Mappings weitergegeben, das über die Rolle VectorFeatureSet in weiteren Verarbeitungsschritten genutzt werden kann, etwa um Vektoren mit einem durch Clusteranalyse reduzierten Merkmalsset zu erstellen.[137] Bei der optionalen Visualisierung werden die gefundenen Clusterzuordnungen farbig hervorgehoben. Da sie in alle Cluster-Komponenten integriert ist, wird die Option hier nicht mit in die Auflistung der Parameter aufgenommen. In der nachfolgenden Aufstellung wird zwischen distanzbasierten und dichtebasierten Verfahren unterschieden (siehe dazu Abschnitt 6.2.7).

A.6.1 Distanzbasierte Verfahren

Das bekannteste distanzbasierte Verfahren ist der K-Means-Algorithmus. Für eine vorgegebene Clusteranzahl *k* werden ausgehend von initial festgelegten Clusterzentren jeweils die ähnlichsten Elemente gruppiert, wobei Ähnlichkeit hier (wie im Word Space Model) über die Distanz der Repräsentationen bestimmt wird. In jedem Durchlauf wird das Clusterzentrum neu bestimmt und anschließend eine erneute Zuordnung der Datenpunkte vorgenommen. Der Algorithmus endet, wenn sich die Schwerpunktberechnung ›stabilisiert‹ (wenn keine neuen Zuordnungen mehr möglich sind) oder wenn eine angegebene maximale Anzahl von Iterationen erreicht wurde. Über die ELKI-API wurden verschiedene Varianten eingebunden, die sich vor allem in der Berechnung der Clusterzentren unterscheiden.

136 Siehe https://elki-project.github.io (Zugriff vom 21.02.2018). Die hier verwendete Version 0.5.5 ist auf den 10.12.2012 signiert.

137 Indem das Attributset zunächst geclustert wird, können die potentiell hunderttausenden Attribute auf eine geringe Zahl abgebildet werden, im Sinne eines Mappings der Merkmale auf Merkmalsklassen.

ELKI K-Means MacQueen Clusterer		
Konsumiert	Vectors, FeatureSet	
Produziert	Clusters, FeatureSet	
Konfiguration	Init	Methode zur Initialisierung (z.B. zufällig)
	Random Seed	Startwert des Zufallszahlengenerators
	No. of Clusters	Anzahl der zu erzeugenden Cluster
	Maximum number of iterations	Anzahl der Iterationen, in denen Cluster-Zentren neu berechnet werden

ELKI K-Means Lloyd Clusterer		
Konsumiert	Vectors, FeatureSet	
Produziert	Clusters, FeatureSet	
Konfiguration	Init	Methode zur Initialisierung (z.B. zufällig)
	Random Seed	Startwert des Zufallszahlengenerators
	No. of Clusters	Anzahl der zu erzeugenden Cluster
	Maximum number of iterations	Anzahl der Iterationen, in denen Cluster-Zentren neu berechnet werden

ELKI K-Medians Lloyd Clusterer		
Konsumiert	Vectors, FeatureSet	
Produziert	Clusters, FeatureSet	
Konfiguration	Init	Methode zur Initialisierung (z.B. zufällig)
	Random Seed	Startwert des Zufallszahlengenerators
	No. of Clusters	Anzahl der zu erzeugenden Cluster
	Maximum number of iterations	Anzahl der Iterationen, in denen Cluster-Zentren neu berechnet werden

ELKI K-Medoids EM Clusterer		
Konsumiert	Vectors, FeatureSet	
Produziert	Clusters, FeatureSet	
Konfiguration	Init	Methode zur Initialisierung (z.B. zufällig)
	Random Seed	Startwert des Zufallszahlengenerators
	No. of Clusters	Anzahl der zu erzeugenden Cluster
	Maximum number of iterations	Anzahl der Iterationen, in denen Cluster-Zentren neu berechnet werden

ELKI K-Medoids PAM Clusterer		
Konsumiert	Vectors, FeatureSet	
Produziert	Clusters, FeatureSet	
Konfiguration	Init	Methode zur Initialisierung (z.B. zufällig)
	Random Seed	Startwert des Zufallszahlengenerators
	No. of Clusters	Anzahl der zu erzeugenden Cluster
	Maximum number of iterations	Anzahl der Iterationen, in denen Cluster-Zentren neu berechnet werden

A.6.2 Dichtebasierte Verfahren

Im Gegensatz zu den oben genannten Algorithmen muss bei dichtebasierten Verfahren die erwartete Clusterzahl nicht vorab angegeben werden. Stattdessen versucht der Algorithmus, innerhalb eines vorgegebenen Radius eine ausreichend große Anzahl von Elementen zu finden (die ebenfalls vorgegeben wird). In dieser Arbeit wird der DBSCAN-Algorithmus eingesetzt, zu dem die beiden anderen hier aufgeführten Verfahren in ihren Konfigurationsmöglichkeiten nur geringfügig abweichen.

ELKI DBSCAN Clusterer		
Konsumiert	Vectors, FeatureSet	
Produziert	Clusters, FeatureSet	
Konfiguration	Epsilon	Maximaler Radius benachbarter Elemente
	MinPts	Mindestgröße für die einzelnen Cluster

ELKI OPTICS Clusterer		
Konsumiert	Vectors, FeatureSet	
Produziert	Clusters, FeatureSet	
Konfiguration	Epsilon	Maximaler Radius benachbarter Elemente
	MinPts	Mindestgröße für die einzelnen Cluster
	Steepness	Schwellwert für den Anstieg der Distanzen innerhalb eines Clusters

ELKI SNN Clusterer		
Konsumiert	Vectors, FeatureSet	
Produziert	Clusters, FeatureSet	
Konfiguration	Epsilon	Grad der Mindest-Dichte in einem Cluster
	MinPts	Mindestgröße für die einzelnen Cluster

A.6.3 ClusterFilter

Die ClusterFilter-Komponente berechnet für jedes Cluster zunächst den Schwerpunkt (Zentroid). Dieser dient als Referenzpunkt für die Ermittlung der ›typischsten‹ Clusterelemente, die anhand ihrer Ähnlichkeit zum Zentroid ausgewählt werden.

Cluster Filter		
Konsumiert	Clusters	
Produziert	Clusters, Vectors	
Konfiguration	nMedoids	Anzahl der Elemente, die behalten werden sollen, ausgehend vom jeweiligen Zentroid
	minSize	Mindestgröße für Cluster

A.7 Visualisierung

Grundlage der Visualisierung ist die Statistik-Software R,[138] die für quantitative Datenanalysen entworfen wurde und standardmäßig eine entsprechende

138 Siehe https://www.r-project.org (Zugriff vom 21.02.2018).

Plotting-Funktion beinhaltet. Neben dieser können in der Visualisierung auch die in R integrierten hierarchischen Clusterverfahren genutzt werden, um die Daten in verschiedenen Baumstrukturen bzw. Dendrogrammen zu organisieren. Die Visualisierung der erzeugten Vektoren und Cluster ist, wie oben beschrieben, zumeist direkt in die Komponenten integriert, die die jeweiligen Strukturen produzieren. Um in den Experimenten eine gezielte und kontrollierte Visualisierung zu ermöglichen, wurde eine separate Plotting-Komponente implementiert, die es zusätzlich gestattet, einzelne Wörter im Plot hervorzuheben.

R Plotter		
Konsumiert	Clusters	
Produziert	Clusters, Vectors	
Konfiguration	Plot type	Art der Visualisierung (z.B. Scatterplot, Dendrogramm, etc.)
	Target	Wort, das im Plot hervorgehoben werden soll

B. Experimente

Experimentdefinitionen werden in Tesla in Form von XML-Dateien hinterlegt (siehe Abb. B.1). Die in Kapitel 7 durchgeführten Experimente sind in der verwendeten Tesla-Distribution hinterlegt. Diese enthält zudem die verwendeten Komponenten und Datenquellen. Im Folgenden werden die in der Arbeit eingesetzten Versuchsaufbauten in ihrer Darstellung im graphischen Editor wiedergegeben.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?> <ns2:tesla_experiment
    createTime="1443959711235" singleton="false" state="1" version="0" id="0"
    xmlns:ns2="http://spinfo.uni-koeln.de/tesla">
    <documentCollections>
        <entry>
            <key>sig_1</key>
            <value numberOfDocuments="1" width="190" posY="16" posX="729"
                localId="sig_1">
                <description>
                        Plain text corpus, part of the Leipzig Corpora Collection
                </description>
                <name>lcc_deu_news_1995_1M</name>
                <producesRoles>ri-1731976018</producesRoles>
                <documentReferences
                    reader="0a601313-a28b-445a-9b71-6c0de0b2f37d"
                    dataSourceId="lcc_deu_news_1995_1M"/>
                <xmlId>a977721d-85c5-486b-b27e-f6695783dc14</xmlId>
            </value>
        </entry>
    </documentCollections>
    <component localId="aa43c1b0-9bec-486f-ac05-8f1d0b013ca5" version="1.0"
        name="TF/IDF" width="150" posY="438" posX="485">
        <className>
                de.uni_koeln.spinfo.tesla.component.statistics.TfIdfCalculator
        </className>
        . . .
        <configuration maximum="1" minimum="1" id="0" category="Reuse Results">
            <description>If false, this component will be executed whenever used
                in an experiment. If true, the annotations produced by this
                component earlier will be reused if the execution prerequesites
                did not change.</description>
            <value>true</value>
        </configuration>
        . . .
</ns2:tesla_experiment>
```

Abbildung B.1: Ausschnitt einer Experimentdefinition in Tesla. Neben den eingesetzten Datenquellen sind hier vor allem die für die einzelnen Komponenten gewählten Parameter dokumentiert, was die Reproduktion der Ergebnisse ermöglicht.

B.1 Kookkurrenzvektoren und Referenzräume (Abschnitt 7.1)

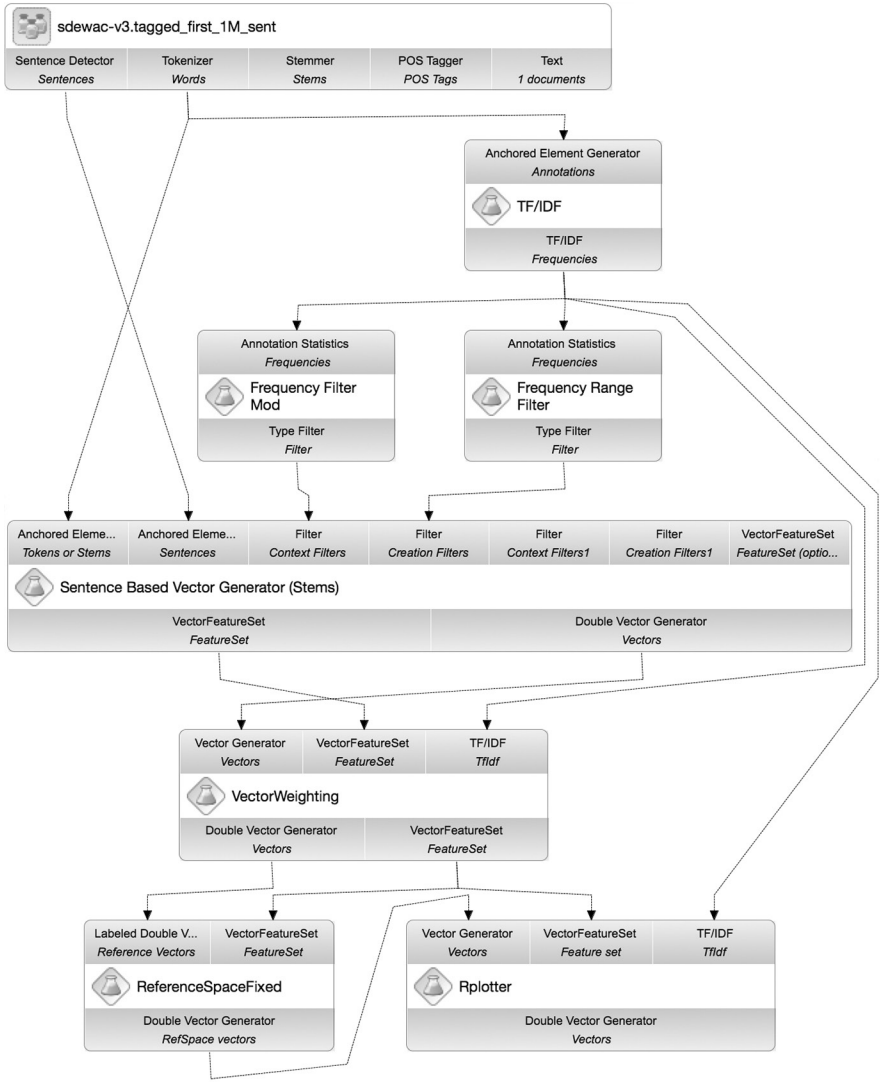


Abbildung B.2: Aufbau des Experiments zu Abb. 7.3. Erstellung eines Referenzraums für das Zielwort *spielen*.

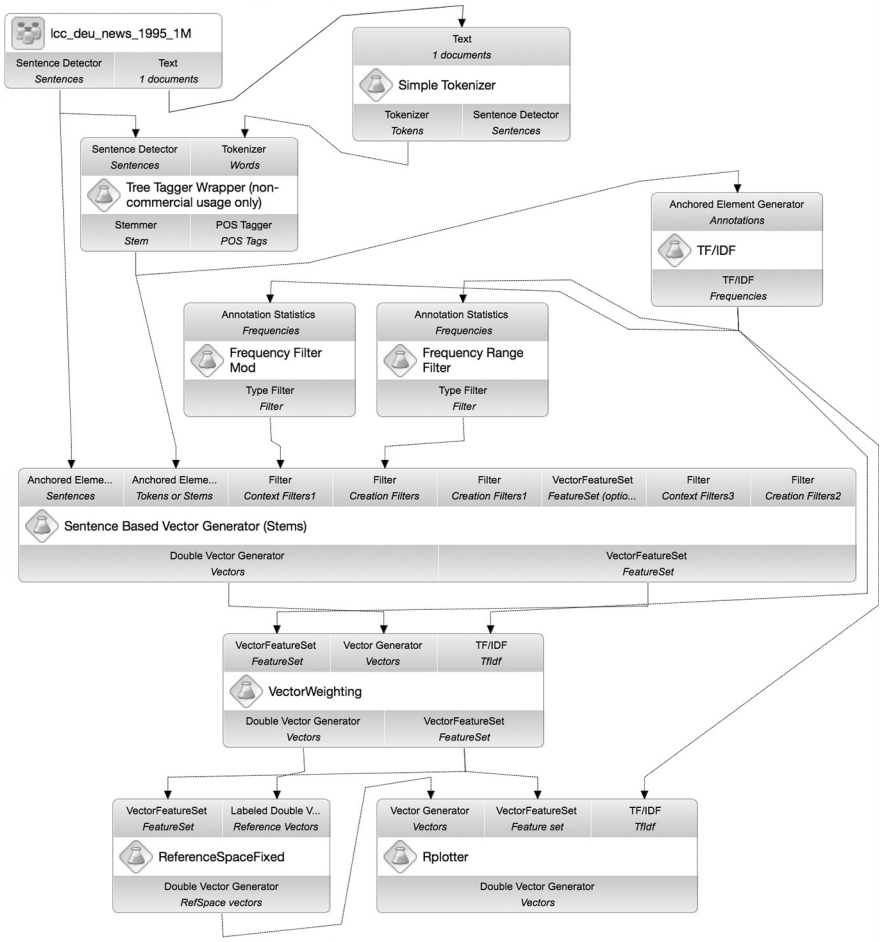


Abbildung B.3: Experiment zu Abb. 7.4 (Plot oben rechts). Erstellung eines Referenzraums für *spielen* auf Grundlage des LCC-Korpus. Die weiteren Plots aus Abb. 7.4 wurden mit dem eben gezeigten Versuchsaufbau (siehe Abb. B.1) erstellt.

B.2 Bedeutungskonstitution in Einzelkontexten (Abschnitt 7.2)

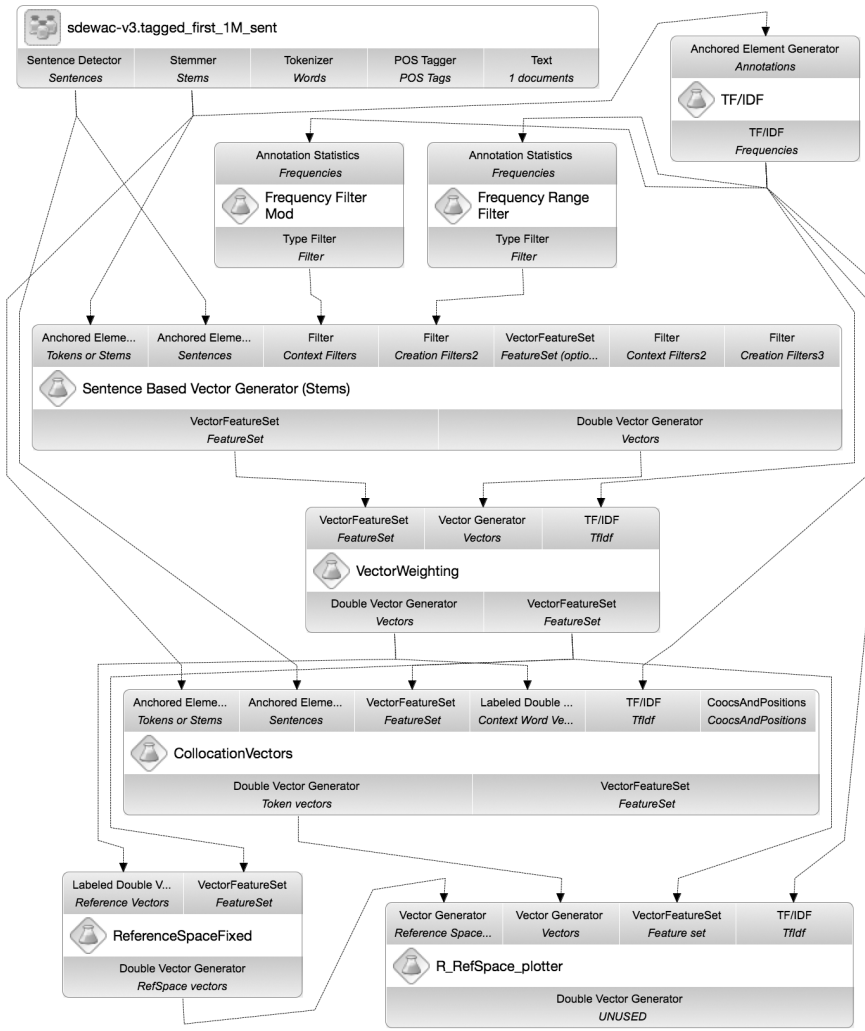


Abbildung B.4: Experiment zu den Abbildungen 7.6, 7.7, und 7.8. Projektion einzelner Kontexte ausgewählter Wörter in ihren jeweiligen Referenzraum.

B.3 Semantische Profile (Abschnitt 7.3)

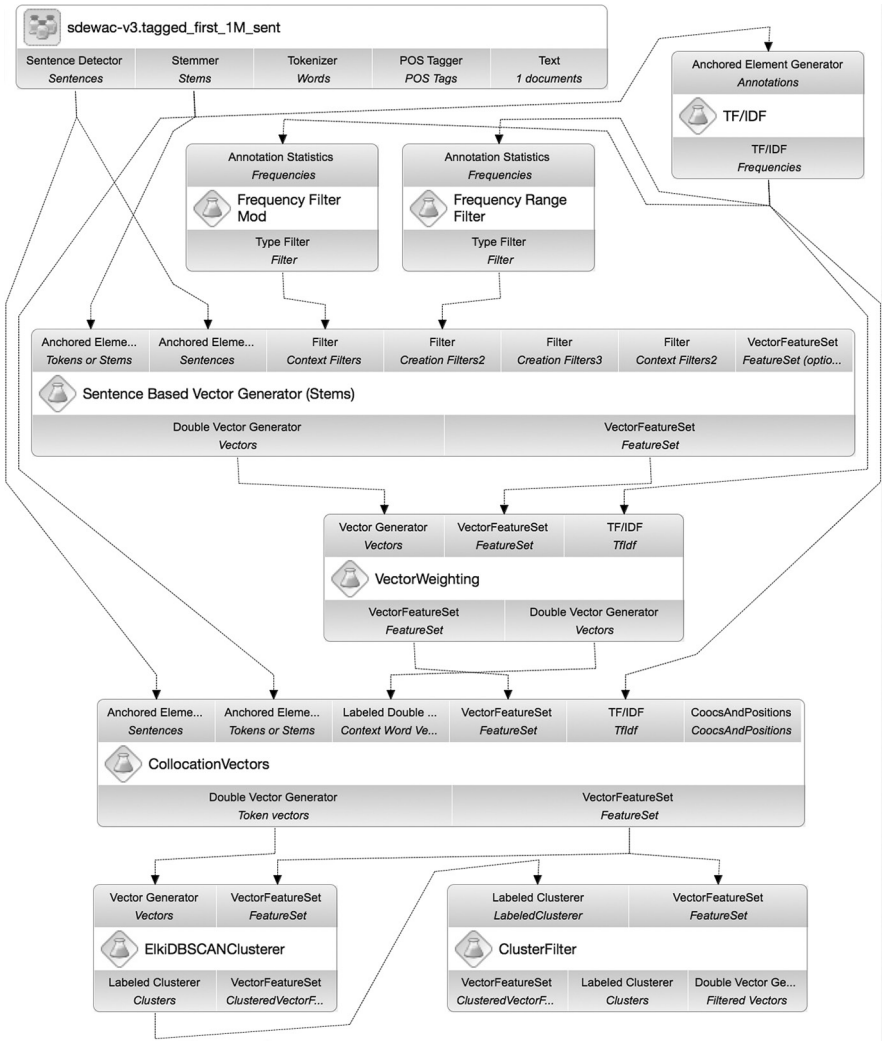


Abbildung B.5: Experiment zu den Abbildungen 7.10, 7.11 und 7.13. Semantische Profile in verschiedener Darstellung auf Grundlage des SdeWaC-Korpus.

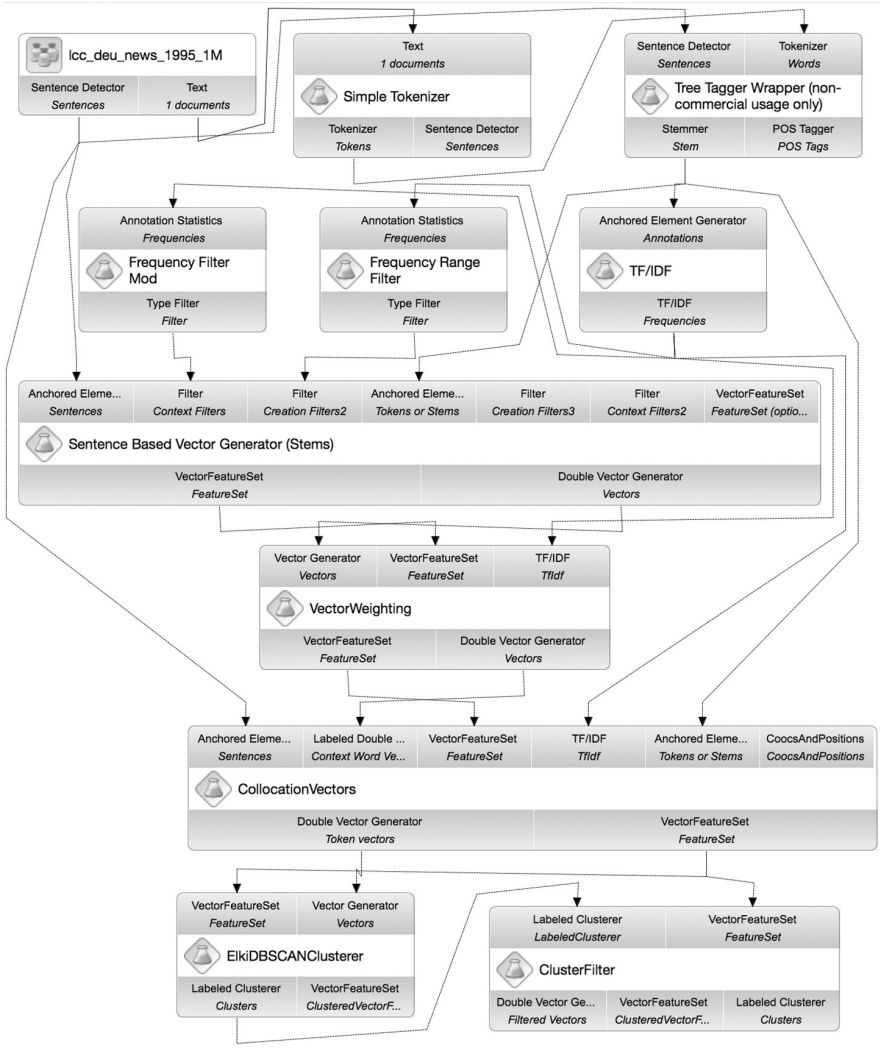


Abbildung B.6: Experiment zu den Abbildungen 7.12 und 7.14. Semantische Profile in verschiedener Darstellung auf Grundlage des LCC-Korpus.

C. Assoziationsmaße

Im Folgenden werden die beiden in dieser Arbeit vorwiegend eingesetzten Assoziationsmaße erläutert. Dies ist zum einen die Pointwise Mutual Information (PMI), die bei der Gewichtung der Vektoren zum Einsatz kommt, zum anderen die Log-Likelihood-Ratio (LLR), die in den Experimenten für die Berechnung der Signifikanz des gemeinsamen Auftretens auf Grundlage der ermittelten Kookkurrenzwerte verwendet wird. Nach Evert (2005) liegt der Berechnung eine sogenannte Kontingenztabelle zugrunde, in der das Auftreten zweier Wörter (*u* und *v*) in Form einer Kreuzklassifikation eingetragen wird (Abb. C.1).

	$V = v$	$V \neq v$
$U = u$	$E_{11} = \frac{R_1 C_1}{N}$	$E_{12} = \frac{R_1 C_2}{N}$
$U \neq u$	$E_{21} = \frac{R_2 C_1}{N}$	$E_{22} = \frac{R_2 C_2}{N}$

	$V = v$	$V \neq v$	
$U = u$	O_{11}	O_{12}	$= R_1$
$U \neq u$	O_{21}	O_{22}	$= R_2$
	$= C_1$	$= C_2$	$= N$

Abbildung C.1: Kontingenztabelle für ein Wortpaar (*u*,*v*), in der die beobachteten Kookkurrenzen (rechte Tabelle) und die zugehörigen Erwartungswerte (linke Tabelle) eingetragen sind (Abbildung nach Evert 2005).

Die rechte Tabelle enthält die tatsächlich beobachteten Häufigkeiten (*O* steht für »observed«): O_{11} ist die Häufigkeit des gemeinsamen Auftretens von *u* und *v*, O_{12} die Frequenz von *u* ohne *v*, O_{21} die Frequenz von *v* ohne *u*, und O_{22} steht für die Anzahl der Wortpaare, die weder *u* noch *v* enthalten. *R* und *C* stehen für die Zeilen- bzw. Spaltensummen, die sich zur Gesamtanzahl aller möglichen Wortpaare (*N*) aufsummieren. Unter Verwendung dieser Werte können die zugehörigen Erwartungswerte errechnet werden (notiert als *E* für »expected«), notiert in der linken Tabelle. Auf dieser Grundlage lässt sich eine Vielzahl von Assoziationsmaßen herleiten (siehe dazu Evert 2005), was eine einfache Übertragung der oftmals komplexen Formeln in Programmcode gestattet. Dies gilt auch für die in dieser Arbeit eingesetzten Maße der PMI und LLR, die im Folgenden kurz erläutert werden.

C.1 Pointwise Mutual Information (PMI)

Die PMI ist ein informationstheoretisch motiviertes Maß, um den Grad der Überlappung zweier Ereignisse (hier: des gemeinsamen Auftretens von Wörtern) zu messen. Im Zusammenhang mit Wortvektoren ist die PMI eines der am weitesten

verbreiteten Maße, in der folgenden Gleichung zunächst wiedergegeben in der Formulierung von Church & Hanks (1990):

$$pmi(x,y) = \log_2 \frac{P(x,y)}{P(x)P(y)}$$

Durch die PMI wird die bedingte Wahrscheinlichkeit des gemeinsamen Auftretens zweier Wörter, notiert als *P(x,y)*, in Relation zu ihren jeweiligen Auftrittswahrscheinlichkeiten gesetzt. Die Umsetzung in dieser Arbeit orientiert sich an Evert (2005), der die PMI über die obigen Kontingenztabellen wie folgt herleitet:

$$PMI = \log \frac{O_{11}}{E_{11}}$$

Wie diese Formulierung deutlich macht, wird durch die PMI das Verhältnis des tatsächlichen gemeinsamen Auftretens gegenüber dem entsprechenden Erwartungswert berechnet.

C.2 Log-Likelihood-Ratio (LLR)

Die LLR nach Dunning (1993), auch als G^2-Test bekannt, ermittelt den Grad der Wahrscheinlichkeit (»likelihood«), ob es sich bei dem gemeinsamen Vorkommen um ein abhängiges oder ein unabhängiges Ereignis handelt. Auch die LLR wird in Evert (2005) unmittelbar über die obige Kontingenztabelle hergeleitet:

$$log-likelihood = 2\sum_{ij} O_{ij} \log \frac{O_{ij}}{E_{ij}}$$

Im Unterschied zur PMI, bei der nur die Kontexte betrachtet werden, in denen mindestens einer der Kookkurrenten auftritt (das heißt alle Fälle bis auf O_{22}), werden bei der LLR sämtliche mögliche Wortpaare in die Berechnung einbezogen. In dieser Arbeit wird eine Implementation aus der Machine-Learning-API Mahout verwendet.[139] Diese stützt sich wesentlich auf einen Blogpost von Ted Dunning aus dem Jahre 2008, in dem er die in Dunning (1993) eingeführte LLR über eine

139 Siehe http://mahout.apache.org (Zugriff vom 21.02.2018); zur konkreten Implementation siehe http://apache.github.io/mahout/0.10.1/docs/mahout-math/org/apache/mahout/math/stats/LogLikelihood.html (Zugriff vom 21.02.2018).

(mehrfache) Berechnung der Entropie erläutert, welche den Erwartungswert bezüglich des Informationsgehalts eines Ereignisses beschreibt.[140]

$$LLR = 2 \cdot N \cdot (H(O_{ij}) - H(R_i) - H(C_i))$$

H bezeichnet hierbei die Shannon-Entropie, definiert als

$$H(X) = -\sum p(x) \ \log \ p(x)$$

Diese wird jeweils für die Zeilensummen (R_1 und R_2), die Spaltensummen (C_1 und C_2) sowie für die gesamte Matrix (das heißt für O_{11}, O_{12}, O_{21} und O_{22}) errechnet. Die verwendete Implementation bietet als zusätzliche Variante auch die Berechnung der root-LLR an, bei der die Wurzel der berechneten LLR zurückgegeben wird. Hier lässt sich die positive gegenüber der negativen Korrelation direkt am Vorzeichen ablesen, da der Wert nur dann positiv ist, wenn er höher ist als der Erwartungswert, sonst negativ.

140 Siehe http://tdunning.blogspot.com/2008/03/surprise-and-coincidence.html (Zugriff vom 21.02.2018). Da mit der LLR im Wesentlichen eine Abwägung zwischen dem Erwartungswert gegenüber dem tatsächlichen Wert vorgenommen wird, hat Dunning seinen Blogpost mit »Surprise and Coincidence« betitelt.

Abbildungsverzeichnis

3.1 Semantische Strukturen als Teilmenge konzeptueller Strukturen
3.2 Schematische Darstellung des Bedeutungspotentials
3.3 Bedeutungskonstitution im *dynamic construal approach*
3.4 Schematische Input-Output-Relation
3.5 Bedeutungskonstitution als informationsverarbeitender Prozess
3.6 Prozessschema der Bedeutungsvariation
4.1 Schematischer Vektorraum
4.2 Term-Dokument-Matrix
4.3 Kookkurenz-Matrix
4.4 Beispiel für die Erstellung von Wortvektoren
4.5 HAL-Matrix
4.6 Kontextvektoren
4.7 Räumliche Nähe im Vektorraum
5.1 Differenzierung des Bedeutungspotentials im WSM
5.2 Eingabeformat für die Bedeutungskonstitution
5.3 Bedeutungskonstitution als Prozess der Transformation
5.4 Kontextuelle Aktivierung im Vektorraum
5.5 Bedeutungskonstitution als zweischrittige Transformation
5.6 Erweiterte Prozessbeschreibung mit Gewichtung
5.7 Bedeutungskonstitution im Vektorraum
5.8 Mehrfache Kontextualisierung im Wortraum
6.1 Beispiel für einen Versuchsaufbau in Tesla
6.2 Die Linguist Perspective
6.3 Ergebnisdarstellung in Tesla
6.4 Schematische Darstellung des Tesla Role System
6.5 Visualisierung durch Scatterplot und Dendrogramm
6.6 Gesamtverteilung mehrdeutiger Einträge in GermaNet
6.7 Einträge mit den meisten Synset-Zuordnungen in GermaNet
6.8 V-A-ambige Wörter in GermaNet
7.1 Workflow zur Erstellung von Wortvektoren in Tesla
7.2 Workflow zur Erstellung von Referenzräumen
7.3 Hierarchische Clusteranalyse des Referenzraums für *spielen*
7.4 Referenzräume in unterschiedlicher Konfiguration
7.5 Workflow für Einzelvorkommen
7.6 Bedeutungskonstitution in ausgewählten Kontexten von *spielen*
7.7 Bedeutungskonstitution in ausgewählten Kontexten von *spielen* (2)
7.8 Mehrfachprojektion ausgewählter Kontexte verschiedener Wörter
7.9 Workflow für die Erstellung semantischer Profile
7.10 Typische Verwendung für *spielen*
7.11 Semantisches Profil von *spielen*
7.12 Semantisches Profil von *spielen* bei veränderter Datengrundlage
7.13 Semantisches Profil von *scharf* bzw. *Krone*
7.14 Bedeutungspotentiale als Bereiche im Vektorraum
B.1 Ausschnitt einer Experimentdefinition

B.2 Experiment zu Abbildung 7.3
B.3 Experiment zu den Abbildungen 7.6, 7.7 und 7.8
B.4 Experiment zu den Abbildungen 7.10, 7.11 und 7.13
B.5 Experiment zu den Abbildungen 7.12 und 7.14
C.1 Kontingenztabelle nach Evert (2005)

Literaturverzeichnis

Achtert, Elke/Sascha Goldhofer/Hans-Peter Kriegel/Erich Schubert/Arthur Zimek (2012): »Evaluation of Clusterings – Metrics and Visual Support«, in: *Proceedings of the IEEE 28th International Conference on Data Engineering (ICDE 2012), Washington D.C., USA, April 1–5, 2012,* Los Alamitos: IEEE Computer Society Press, 1285–1288, DOI: https://doi.org/10.1109/ICDE.2012.128 (Zugriff vom 21.02.2018).

Almuhareb, Abdulrahman/Massimo Poesio (2004): »Attribute-Based and Value-Based Clustering: An Evaluation«, in: *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP 2004), Barcelona, Spain, July 25–26, 2004,* 158–165, URL: http://www.aclweb.org/anthology/W04-3221 (Zugriff vom 21.02.2018).

Ankerst, Mihael/Markus M. Breunig/Hans-Peter Kriegel/Jörg Sander (1999): »OPTICS: Ordering Points to Identify the Clustering Structure«, in: *Proceedings of the 1999 ACM SIGMOD International Conference on Management of Data (SIGMOD '99), Philadelphia, PA, USA, June 1–3, 1999,* New York, 49–60, DOI: http://dx.doi.org/10.1145/304182.304187 (Zugriff vom 21.02.2018).

Baroni, Marco/Silvia Bernardini/Adriano Ferraresi/Eros Zanchetta (2009): »The WaCky Wide Web: A collection of very large linguistically processed webcrawled corpora«, in: *Language Resources and Evaluation* 43/3, 209–226,[10.02.2009], DOI: https://doi.org/10.1007/s10579-009-9081-4 (Zugriff vom 21.02.2018).

Baroni, Marco/Adam Kilgarriff (2006): »Large Linguistically-processed Web Corpora for Multiple Languages«, in: *Proceedings of the Eleventh Conference of the European Chapter of the Association for Computational Linguistics: Posters & Demonstrations (EACL '06), Trento, Italy, April 5–6, 2006,* Stroudsburg, 87–90, URL: http://dl.acm.org/citation.cfm?id=1608974.1608976 (Zugriff vom 21.02.2018).

Bellman, Richard (1961): *Adaptive Control Processes: A Guided Tour* (Princeton Legacy Library), Princeton: Princeton University Press.

Bierwisch, Manfred (1983): »Semantische und konzeptuelle Repräsentationen lexikalischer Einheiten«, in: Rudolf Růžička/Wolfgang Motsch (Hg.), *Untersuchungen zur Semantik* (studia grammatica, 22), Berlin: Akademie-Verlag, 61–99.

Bierwisch, Manfred/Ewald Lang (Hg.) (1987): *Grammatische und konzeptuelle Aspekte von Dimensionsadjektiven*, Berlin: Akademie-Verlag.

Bierwisch, Manfred/Ewald Lang (Hg.) (1989): *Dimensional Adjectives: Grammatical Structure and Conceptual Interpretation* (Springer Series in Language and Communication), Berlin: Springer.

Bird, Steven/Mark Liberman (2001): »A Formal Framework for Linguistic Annotation«, in: *Speech Communication* 33, 23–60, DOI: https://doi.org/10.1016/S0167-6393(00)00068-6 (Zugriff vom 21.02.2018).

Burgess, Curt (1998): »From simple associations to the building blocks of language: Modeling meaning in memory with the HAL model«, in: *Behavior Research Methods, Instruments, & Computers* 30/2, 188–198.

Burgess, Curt/Kay Livesay/Kevin Lund (1998): »Explorations in context space: Words, sentences, discourse«, in: *Discourse Processes* 25/2–3, 211–257, DOI: http://dx.doi.org/10.1080/01638539809545027 (Zugriff vom 21.02.2018).

Chomsky, Noam (1957): *Syntactic Structures* (Janua Linguarum, 4), The Hague: Mouton.

Chomsky, Noam (1959): »A Review of B.F. Skinner, Verbal Behavior«, in: *Language* 35/1, 26–58, DOI: http://dx.doi.org/10.2307/411334 (Zugriff vom 21.02.2018).

Chomsky, Noam (1965): *Aspects of the Theory of Syntax* (Special technical report, 11), Cambridge, Mass.: The MIT Press.

Chomsky, Noam (1981): *Lectures on Government and Binding* (Studies in Generative Grammar, 9), Dordrecht: Foris Publications.

Church, Kenneth Ward/Patrick Hanks (1990): »Word Association Norms, Mutual Information, and Lexicography«, in: *Computational Linguistics* 16/1, 22–29, URL: http://www.aclweb.org/anthology/P89-1010.pdf (Zugriff vom 21.02.2018).

Croft, William/David A. Cruse (2004): *Cognitive Linguistics* (Cambridge Textbooks in Linguistics), Cambridge: Cambridge University Press, [06.2012], DOI: https://doi.org/10.1017/CBO9780511803864 (Zugriff vom 21.02.2018).

Cruse, David A. (1986): *Lexical Semantics* (Cambridge Textbooks in Linguistics), Cambridge: Cambridge University Press.

Cruse, David A. ([2004] 2011): *Meaning in Language: An Introduction to Semantics and Pragmatics* (Oxford Textbooks in Linguistics), 3. Aufl., Oxford: Oxford University Press.

Cunningham, Hamish (2000): »Software architecture for language engineering«, Dissertation, University of Sheffield, URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.12.325&rep=rep1&type=pdf (Zugriff vom 21.02.2018).

Dagan, Ido/Kenneth W. Church/William. A. Gale (1993a): »Robust bilingual word alignment for machine aided translation«, in: *Proceedings of the Workshop on Very Large Corpora: Academic and Industrial Perspectives, Columbus, Ohio, USA, June 22, 1993*, 1–8, URL: https://aclanthology.coli.uni-saarland.de/pdf/W/W93/W93-0300.pdf (Zugriff vom 21.02.2018).

Dagan, Ido/Shaul Marcus/Shaul Markovitch (1993b): »Contextual word similarity and estimation from sparse data«, in: *Proceedings of the 31st annual meeting on Association for Computational Linguistics, Columbus, Ohio, USA, June 22–26, 1993*, Stroudsburg, 164–171, DOI: https://doi.org/10.3115/981574.981596 (Zugriff vom 21.02.2018).

Deerwester, Scott/Susan T. Dumais/George W. Furnas/Thomas K. Landauer/Richard Harshman (1990): »Indexing by Latent Semantic Analysis«, in: *Journal of the American Society for Information Science* 41/6, 391–407, DOI: https://doi.org/10.1002/(SICI)1097-4571(199009)41:6<391::AID-ASI1>3.0.CO;2-9 (Zugriff vom 21.02.2018).

Dumais, Susan T./George W. Furnas/Thomas. K. Landauer/Scott Deerwester/Richard Harshman (1988): »Using Latent Semantic Analysis to Improve Access to Textual Information«, in: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, *Washington, D.C., USA, May 15–19, 1988,* New York, 281–285, DOI: https://doi.org/10.1145/57167.57214 (Zugriff vom 21.02.2018).

Dunning, Ted (1993): »Accurate Methods for the Statistics of Surprise and Coincidence«, in: *Computational Linguistics* 19/1, 61–74, URL: http://aclweb.org/anthology/J93-1003 (Zugriff vom 21.02.2018).

Ertöz, Levent/Michael Steinbach/Vipin Kumar (2003): »Finding clusters of different sizes, shapes, and densities in noisy, high dimensional data«, in: *Proceedings of the Third SIAM International Conference on Data Mining, San Francisco, USA, May 1–3, 2003,* Philadelphia: SIAM, Society for Industrial and Applied Mathematics, 47–58, https://doi.org/10.1137/1.9781611972733.5 (Zugriff vom 21.02.2018).

Ester, Martin/Hans-Peter Kriegel/Jörg Sander/Xiaowei Xu (1996): »A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise«, in: *Proceedings of Second International Conference on Knowledge Discovery and Data Mining (KDD-96),*

Portland, USA, August 2–4, 1996, Palo Alto: The AAAI Press, 226–231, URL: https://www.aaai.org/Papers/KDD/1996/KDD96-037.pdf (Zugriff vom 21.02.2018).

Evans, Vyvyan/Melanie Green (2006): *Cognitive Linguistics: An Introduction*, Edinburgh: Edinburgh University Press, URL: http://npu.edu.ua/!e-book/book/djvu/A/iif_kgpm_Cognitive%20Linguistics%20An%20Introduction.pdf (Zugriff vom 21.02.2018).

Evert, Stefan (2005): »The Statistics of Word Cooccurrences: Word Pairs and Collocations«, Dissertation, Universität Stuttgart, DOI: http://dx.doi.org/10.18419/opus-2556 (Zugriff vom 21.02.2018).

Fauconnier, Gilles (1994): *Mental Spaces: Aspects of Meaning Construction in Natural Language,* Cambridge: Cambridge University Press, [01.2010], DOI: https://doi.org/10.1017/CBO9780511624582 (Zugriff vom 21.02.2018).

Ferrucci, David/Adam Lally (2003): »Accelerating Corporate Research in the Development, Application and Deployment of Human Language Technologies«, in: *Proceedings of the HLT-NAACL 2003 Workshop on Software Engineering and Architectures for Language Technology Systems (SEALTS '03), Edmonton, Canada, May 31, 2003*, Stroudsburg, 67–74, DOI: https://dx.doi.org/10.3115/1119226.1119236 (Zugriff vom 21.02.2018).

Ferrucci, David/Adam Lally (2004): »UIMA: An Architectural Approach to Unstructured Information Processing in the Corporate Research Environment«, in: *Natural Language Engineering* 10/3–4, 327–348, DOI: https://dx.doi.org/10.1017/S1351324904003523 (Zugriff vom 21.02.2018).

Fillmore, Charles J. (1976): »Frame Semantics and the Nature of Language«, in: *Annals of the New York Academy of Sciences* 280/1, 20–32, DOI: https://doi.org/10.1111%2Fj.1749-6632.1976.tb25467.x (Zugriff vom 21.02.2018).

Fillmore, Charles J. (1982): »Frame Semantics«, in: The Linguistic Society of Korea (Hg.), *Linguistics in the Morning Calm,* Seoul: Hanshin Publishing Co., 111–137, DOI: https://doi.org/10.1016/B0-08-044854-2/00424-7 (Zugriff vom 21.02.2018).

Firth, John Rupert (1957): *Papers in Linguistics, 1934–1951,* London/New York: Oxford University Press.

Gärdenfors, Peter (2004): *Conceptual Spaces: The Geometry of Thought,* Cambridge, Mass.: The MIT Press.

Gärdenfors, Peter (2014): *The Geometry of Meaning: Semantics Based on Conceptual Spaces,* Cambridge, Mass.: The MIT Press.

Geeraerts, Dirk (1993): »Vagueness's puzzles, polysemy's vagaries'«, in: *Cognitive Linguistics* 4/3, 223–272, DOI: https://doi.org/10.1515/cogl.1993.4.3.223 (Zugriff vom 21.02.2018).

Geeraerts, Dirk (2006a): »A Rough Guide to Cognitive Linguistics«, in: Dirk Geeraerts (Hg.), *Cognitive Linguistics: Basic Readings* (Cognitive Linguistics Research, 34), Berlin/New York: De Gruyter Mouton, 1–28, DOI: https://doi.org/10.1515/9783110199901.1 (Zugriff vom 21.02.2018).

Geeraerts, Dirk (Hg.) (2006b): *Cognitive Linguistics: Basic Readings* (Cognitive Linguistics Research,34),Berlin/NewYork:DeGruyterMouton,DOI:https://doi.org/10.1515/9783110199901 (Zugriff vom 21.02.2018).

Glynn, Dylan/Kerstin Fischer (Hg.) (2010): *Quantitative Methods in Cognitive Semantics: Corpus-Driven Approaches* (Cognitive Linguistics Research, 46). Berlin/New York: De Gruyter Mouton, URL: http://www.dsglynn.univ-paris8.fr/Glynn_&_Fischer_2010_Quantitative_Methods_in_Cognitive_Semantics.pdf (Zugriff vom 21.02.2018).

Goldhahn, Dirk/Thomas Eckart/Uwe Quasthoff (2012): »Building Large Monolingual Dictionaries at the Leipzig Corpora Collection: From 100 to 200 Languages«, in: *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*,

Istanbul, Turkey, May 21–27, 2012, Paris: European Language Resources Association (ELRA), 759–765, URL: http://www.lrec-conf.org/proceedings/lrec2012/pdf/327_Paper.pdf (Zugriff vom 21.02.2018).

Grefenstette, Gregory (1994): *Explorations in Automatic Thesaurus Discovery*, Norwell, Mass.: Kluwer.

Halliday, Michael A. K. (1973): *Explorations in the Functions of Language,* London: Edward Arnold.

Halliday, Michael A. K. (1978): *Language as Social Semiotic: The Social Interpretation of Language and Meaning,* Baltimore: University Park Press.

Hamp, Birgit/Helmut Feldweg (1997): »GermaNet – a Lexical-Semantic Net for German«, in: *Proceedings of ACL/EACL workshop on Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications, Madrid, Spanien, July 12, 1997*, Madrid: ACL, URL: http://www.aclweb.org/anthology/W97-0802 (Zugriff vom 21.02.2018).

Harris, Zellig (1954): »Distributional structure«, in: *Word* 10/2–3, 146–162, DOI: https://doi.org/10.1080/00437956.1954.11659520 (Zugriff vom 21.02.2018).

Harris, Zellig (1968): *Mathematical structures of language* (Interscience Tracts in Pure and Applied Mathematics, 21), New York: Interscience Publishers John Wiley & Sons.

Hearst, Marti A. (1992): »Automatic Acquisition of Hyponyms from Large Text Corpora«, in: *Proceedings of the 14th Conference on Computational Linguistics (COLING '92), Nantes, France, August 23–28, 1992*, Stroudsburg, 539–545. DOI: https://doi.org/10.3115/992133.992154 (Zugriff vom 21.02.2018).

Henrich, Verena/Erhard Hinrichs (2010): »GernEdiT - The GermaNet Editing Tool«, in: *Proceedings of the ACL 2010 System Demonstrations, Uppsala, Sweden, July 13, 2010*, Stroudsburg, 19–24, URL: http://www.aclweb.org/anthology/P10-4004 (Zugriff vom 21.02.2018).

Hermes, Jürgen (2012): »Textprozessierung – Design und Applikation«, Dissertation, Universität zu Köln, [22.02.2012], Permalink: http://nbn-resolving.org/urn:nbn:de:hbz:38-45617 (Zugriff vom 21.02.2018).

Hermes, Jürgen/Stephan Schwiebert (2010): »Classification of Text Processing Components: The Tesla Role System«, in: Andreas Fink/Berthold Lausen/Wilfried Seidel/Alfred Ultsch (Hg.), *Advances in Data Analysis, Data Handling and Business Intelligence. Studies in Classification, Data Analysis, and Knowledge Organization*, Berlin/Heidelberg: Springer, 285–294, DOI: https://doi.org/10.1007/978-3-642-01044-6_26 (Zugriff vom 21.02.2018).

Heylen, Kris/José Tummers/Dirk Geeraerts (2008): »Methodological issues in corpus-based Cognitive Linguistics«, in: Gitte Kristiansen & René Dirven (Hg.), *Cognitive Sociolinguistics: Language Variation, Cultural Models, Social Systems* (Cognitive Linguistics Research, 39), Berlin: De Gruyter Mouton, 91–128, DOI: https://doi.org/10.1515/9783110199154.2.91 (Zugriff vom 21.02.2018).

Heylen, Kris/Thomas Wielfaert/Dirk Speelman/Dirk Geeraerts (2015): »Monitoring polysemy: Word space models as a tool for large-scale lexical semantic analysis«, in: *Lingua* 157, 153–172, DOI: https://doi.org/10.1016/j.lingua.2014.12.001 (Zugriff vom 21.02.2018).

Kanerva, Pentti (2009): »Hyperdimensional Computing: An Introduction to Computing in Distributed Representation with High-Dimensional Random Vectors«, in: *Cognitive Computation* 1/2, 139–159, DOI: https://doi.org/10.1007/s12559-009-9009-8 (Zugriff vom 21.02.2018).

Karlgren, Jussi/Magnus Sahlgren (2001): »From Words to Understanding«, in: Yoshinori Uesaka, Pentti Kanerva & Hideki Asoh (Hg.), *Foundations of Real-world Intelligence* (CSLI

lecture notes, 125), Stanford: CSLI Publications, 294–308, URL: http://soda.swedish-ict.se/131/1/KarlgrenSahlgren2001.pdf (Zugriff vom 21.02.2018).

Lakoff, George (1987): *Women, fire, and dangerous things: what categories reveal about the mind,* Chicago: University of Chicago Press.

Lakoff, George/Mark Johnson (1980): *Metaphors we live by,* Chicago: University of Chicago Press.

Lakoff, George/Mark Johnson (1999): *Philosophy in the Flesh: The Embodied Mind and Its Challenge to Western Thought,* New York: Basic Books.

Landauer, Thomas K./Susan T. Dumais (1997): »A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge«, in: *Psychological review* 104/2, 211–240, DOI: http://dx.doi.org/10.1037/0033-295X.104.2.211 (Zugriff vom 21.02.2018).

Langacker, Ronald (1987): *Foundations of Cognitive Grammar, Volume 1: Theoretical prerequisites,* Stanford: Stanford University Press.

Langacker, Ronald (1991): *Foundations of Cognitive Grammar, Volume 2: Descriptive Application,* Stanford: Stanford University Press.

Langacker, Ronald (2008): *Cognitive Grammar: A Basic Introduction,* New York: Oxford University Press.

Lenci, Alessandro (2008): »Distributional semantics in linguistic and cognitive research«, in: *Italian Journal of Linguistics* 20/1, 1–30, URL: http://www.italian-journal-linguistics.com/wp-content/uploads/ALenci.pdf (Zugriff vom 21.02.2018).

Levy, Joseph P./John A. Bullinaria (2001): »Learning Lexical Properties from Word Usage Patterns: Which Context Words Should be Used?«, in: Robert M. French & Jacques P. Sougné (Hg.), *Connectionist Models of Learning, Development and Evolution. Perspectives in Neural Computing,* London: Springer, 273–282, DOI: https://doi.org/10.1007/978-1-4471-0281-6_27 (Zugriff vom 21.02.2018).

Lloyd, Stuart P. (1982): »Least Squares Quantization in PCM«, in: *IEEE Transactions on Information Theory* 28, 129–137, DOI: https://doi.org/10.1109/TIT.1982.1056489 (Zugriff vom 21.02.2018).

Lund, Kevin/Curt Burgess (1996): »Producing High-Dimensional Semantic Spaces from Lexical Co-Occurrence«, in: *Behavior Research Methods Instruments and Computers* 28/2, 203–208, DOI: https://doi.org/10.3758/BF03204766 (Zugriff vom 21.02.2018).

Lyons, John ([1971] 1995): *Einführung in die moderne Linguistik*, 8. Aufl., München: C.H. Beck.

MacQueen, James (1967): »Some methods for classification and analysis of multivariate observations«, in: Lucien Marie Le Cam/Jerzy Neyman (Hg.), *Fifth Berkeley Symposium on Mathematical Statistics and Probability,* Berkeley: University of California Press, 281–297.

Manning, Christopher D./Prabhakar Raghavan/Hinrich Schütze (2008): *Introduction to Information Retrieval*, New York: Cambridge University Press. URL: https://nlp.stanford.edu/IR-book/ (Zugriff vom 21.02.2018).

Manning, Christopher D./Hinrich Schütze (1999): *Foundations of Statistical Natural Language Processing,* Cambridge, Mass.: The MIT Press.

Marr, David (1982): *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information,* New York: Henry Holt and Co.

McEnery, Tony/Andrew Wilson (2001): *Corpus Linguistics: An Introduction,* Edinburgh: Edinburgh University Press.

Miller, George A./Walter G. Charles (1991): »Contextual Correlates of Semantic Similarity«, in: *Language and Cognitive Processes* 6/1, 1–28, DOI: https://doi.org/10.1080/01690969108406936 (Zugriff vom 21.02.2018).

Moore, Terence/Christine Carling (1982): *Understanding Language: Towards a Post-Chomskyan Linguistics*, London: Palgrave Macmillan, DOI: https://doi.org/10.1007/978-1-349-16895-8 (Zugriff vom 21.02.2018).

Padó, Sebastian/Mirella Lapata (2007): »Dependency-Based Construction of Semantic Space Models«, in: *Computational Linguistics* 33/2, 161–199, DOI: https://doi.org/10.1162/coli.2007.33.2.161 (Zugriff vom 21.02.2018).

Peirsman, Yves/Kris Heylen/Dirk Geeraerts (2010): »Applying word space models to sociolinguistics. Religion names before and after 9/11«, in: Dirk Geeraerts/Gitta Kristiansen/Yves Peirsman (Hg.), *Advances in Cognitive Sociolinguistics* (Cognitive Linguistics Research, 45), Berlin: De Gruyter Mouton, 111–137, DOI: https://doi.org/10.1515/9783110226461.111 (Zugriff vom 21.02.2018).

Peirsman, Yves/Kris Heylen/Dirk Speelman (2008): »Putting things in order. First and second order context models for the calculation of semantic similarity«, in: *Actes JADT 2008: 9es journées internationales d'analyse statistique des données textuelles, Lyon, 12–14 mars 2008/ Proceedings JADT 2008: 9th international conference on textual data statistical analysis, Lyon, March 12–14, 2008,* Lyon: Presses universitaires de Lyon, 907–916, URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.477.3917&rep=rep1&type=pdf (Zugriff vom 21.02.2018).

Pennacchiotti, Marco/Patrick Pantel (2009): »Entity extraction via ensemble semantics«, in: *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP '09), Singapore, August 06–07, 2009*, Stroudsburg, 238–247, URL: http://wmmks.csie.ncku.edu.tw/ACL-IJCNLP-2009/EMNLP/pdf/EMNLP025.pdf (Zugriff vom 21.02.2018).

Porter, Michael (1980): »An algorithm for suffix stripping«, in: *Program* 14/3, 130–137.

Pustejovsky, James (1998): *The Generative Lexicon,* Cambridge, Mass.: The MIT Press.

Pustejovsky, James/Elisabetta Jezek (2008): »Semantic Coercion in Language: Beyond Distributional Analysis«, in: *Italian Journal of Linguistics 20/1*, 181–214, URL: http://www.italian-journal-linguistics.com/wp-content/uploads/Pustejovsky_Jezek.pdf (Zugriff vom 21.02.2018).

Quasthoff, Uwe/Matthias Richter/Christian Biemann (2006): »Corpus Portal for Search in Monolingual Corpora«, in: *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC 2006), Genova, Italy, May 24–26, 2006,* Paris: ELRA, 1799–1802, URL: https://pdfs.semanticscholar.org/065c/4c163895e55f319507ae43b1cf85b4ab966f.pdf (Zugriff vom 21.02.2018).

Richter, Michael/Jürgen Hermes/Claes Neuefeind (2015), »Automatic Induction of German Aspectual Verb Classes in a Distributional Framework«, in: *Proceedings of the International Conference of the German Society for Computational Linguistics and Language Technology*, GSCL 2015, Sept. 30–Oct. 2, 2015, Duisburg, 122–129, URL: http://www.gscl.org/proceedings/2015/GSCL-201518.pdf (Zugriff vom 21.02.2018).

Rickheit, Gert/Sabine Weiss/Hans-Jürgen Eikmeyer (2010): *Kognitive Linguistik: Theorien, Modelle, Methoden* (UTB, 3408), Tübingen/Basel: Francke.

Rieger, Burghard (1977), »Bedeutungskonstitution. Einige Bemerkungen zur semiotischen Problematik eines linguistischen Problems«, in: *Zeitschrift für Literaturwissenschaft und Linguistik* 27/28, 55–68, URL: https://www.researchgate.net/profile/Burghard_Rieger/publication/245582945_Bedeutungskonstitution_Bemerkungen_zur_semiotischen_Proble-matik_eines_linguistischen_Problems/links/561943c008aea80367203050/Bedeutungskonstitution-Bemerkungen-zur-semiotischen-Proble-matik-eines-linguistischen-Problems.pdf (Zugriff vom 21.02.2018).

Rieger, Burghard (1980): »Fuzzy Word Meaning Analysis and Representation in Linguistic Semantics, an Empirical Approach to the Reconstruction of Lexical Meanings in East- and West-German Newspaper Texts«, in: *Proceedings of the 8th International Conference on Computational Linguistics, COLING '80, Tokyo, Sept. 30–Oct. 4, 1980*, Tokyo, 76–84, DOI: https://doi.org/10.3115/990174.990188 (Zugriff vom 21.02.2018).

Rieger, Burghard (1985): *Dynamik in der Bedeutungskonstitution: Beiträge des Deutschen Germanistentags von 3. bis 6. Oktober, 1982 in Aachen* (Papiere zur Textlinguistik), Hamburg: Helmut Buske.

Rieger, Burghard (1989): *Unscharfe Semantik: die empirische Analyse, quantitative Beschreibung, formale Repräsentation und prozedurale Modellierung vager Wortbedeutungen in Texten,* Frankfurt am Main: Lang.

Rolshoven, Jürgen/Stephan Schwiebert (2007): »Evidenzprozesse: Korpora, Kompression und Musterbildung«, in Claudia M. Riehl/Astrid Rothe (Hg.), *Was ist linguistische Evidenz? Kolloquium des Zentrums Sprachenvielfalt und Mehrsprachigkeit, November 2006*, Aachen: Shaker, 91–108.

Rosch, Eleanor (1975): »Cognitive reference points«, in: *Cognitive Psychology* 7/4, 532–547, DOI: https://doi.org/10.1016/0010-0285(75)90021-3 (Zugriff vom 21.02.2018).

Rosch, Eleanor (1978): »Principles of categorization«, in: Eleanor Rosch/Barbara B. Lloyd (Hg.), *Cognition and Categorization,* Hillsdale: Lawrence Erlbaum, URL: http://commonweb.unifr.ch/artsdean/pub/gestens/f/as/files/4610/9778_083247.pdf (Zugriff vom 21.02.2018).

Rubenstein, Herbert/John B. Goodenough (1965): »Contextual Correlates of Synonymy«, in: *Communications of the ACM* 8/10, 627–633, DOI: https://doi.org/10.1145/365628.365657 (Zugriff vom 21.02.2018).

Ruge, Gerda (1992): »Experiments on linguistically-based term associations«, in: *Information Processing Management* 28/3, 317–332, DOI: https://doi.org/10.1016/0306-4573(92)90078-E (Zugriff vom 21.02.2018).

Ruge, Gerda (1995): *Wortbedeutung und Termassoziation: Methoden zur automatischen semantischen Klassifikation* (Sprache und Computer, 14), Hildesheim: Olms.

Russell, Stuart/Peter Norvig ([1995] 2012): *Künstliche Intelligenz: ein moderner Ansatz*, 3. Aufl., München: Pearson.

Sahlgren, Magnus (2005): »An Introduction to Random Indexing«, in: *Methods and Applications of Semantic Indexing, Workshop at the 7th International Conference on Terminology and Knowledge Engineering (TKE 2005), Kopenhagen, Dänemark, August 16, 2005,* URL: http://soda.swedish-ict.se/221/1/RI_intro.pdf (Zugriff vom 21.02.2018).

Sahlgren, Magnus (2006): »The Word-Space Model: Using Distributional Analysis to Represent Syntagmatic and Paradigmatic Relations Between Words in High-dimensional Vector Spaces«, SICS dissertation series, Stockholm University, URL: http://eprints.sics.se/437/1/TheWordSpaceModel.pdf (Zugriff vom 21.02.2018).

Sahlgren, Magnus (2008): »The Distributional Hypothesis«, in: *Italian Journal of Linguistics* 20/1, 33–54, URL: http://www.italian-journal-linguistics.com/wp-content/uploads/Sahlgren.pdf (Zugriff vom 21.02.2018).

Saitou, Naruya/Masatoshi Nei (1987): »The neighbor-joining method: a new method for reconstructing phylogenetic trees«, in: *Molecular biology and evolution* 4/4, 406–425, DOI: https://doi.org/10.1093/oxfordjournals.molbev.a040454 (Zugriff vom 21.02.2018).

Salton, Gerald (1971): *The SMART Retrieval System – Experiments in Automatic Document Processing,* Englewood Cliffs: Prentice-Hall.

Salton, Gerald/Michael McGill (1983): *Introduction to Modern Information Retrieval*, New York: McGraw-Hill.

Salton, Gerald/Anita Wong/Chung-Shu Yang (1975): »A Vector Space Model for Automatic Indexing«, in: *Communications of the ACM* 18/11, 613–620, DOI: https://doi.org/10.1145/361219.361220 (Zugriff vom 21.02.2018).

Saussure, Ferdinand de ([1931] 1967): *Grundfragen der allgemeinen Sprachwissenschaft*, 2. Aufl., Berlin: Walter de Gruyter.

Schmid, Helmut (1994): »Probabilistic Part-of-Speech Tagging using Decision Trees«, in: *Proceedings of the International Conference on New Methods in Language Processing, Manchester, UK, 1994,* Manchester, 44–49, URL: http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/data/tree-tagger1.pdf (Zugriff vom 21.02.2018).

Schmid, Helmut (2000): »Unsupervised Learning of Period Disambiguation for Tokenisation«, Technical report, Universität Stuttgart, URL: http://www.cis.uni-muenchen.de/~schmid/papers/tokeniser.pdf (Zugriff vom 21.02.2018).

Schütze, Hinrich (1992): »Dimensions of Meaning«, in: *Proceedings of the 1992 ACM/IEEE Conference on Supercomputing (Supercomputing '92), Minneapolis, Minnesota, USA, November 16–20, 1992*, Los Alamitos: IEEE Computer Society Press, 787–796, DOI: https://doi.org/10.1109/SUPERC.1992.236684 (Zugriff vom 21.02.2018).

Schütze, Hinrich (1993): »Word Space«, in: Stephen J. Hanson/Jack D. Cowan/C. Lee Giles (Hg.), *Advances in Neural Information Processing Systems 5 (NIPS 1992),* San Francisco: Morgan-Kaufmann, 895–902, URL: https://papers.nips.cc/paper/603-word-space.pdf (Zugriff vom 21.02.2018).

Schütze, Hinrich (1998): »Automatic Word Sense Discrimination«, in: *Computational Linguistics* 24/1, 97–123, URL: https://aclanthology.info/pdf/J/J98/J98-1004.pdf (Zugriff vom 21.02.2018).

Schütze, Hinrich/Jan O. Pedersen (1997): »A Cooccurrence-based Thesaurus and Two Applications to Information Retrieval«, in: *Information Process and Management* 33/3, 307–318, DOI: https://doi.org/10.1016/S0306-4573(96)00068-4 (Zugriff vom 21.02.2018).

Schwarz, Monika ([1992] 2008): *Einführung in die Kognitive Linguistik* (UTB,1636), 3. Aufl., Tübingen/Basel: Francke.

Schwiebert, Stephan (2012): »Tesla – ein virtuelles Labor für experimentelle Computer- und Korpuslinguistik«, Dissertation, Universität zu Köln, [09.03.2012], Permalink: http://nbn-resolving.org/urn:nbn:de:hbz:38-45716 (Zugriff vom 21.02.2018).

Sinclair, John (1991): *Corpus, Concordance, Collocation* (Describing English Language), Oxford: Oxford University Press.

Skinner, Burrhus Frederic (1957): *Verbal Behavior* (Century Psychology Series), New York: Appleton-Century-Crofts.

Sokal, Robert R./Charles D. Michener (1958): »A statistical method for evaluating systematic relationships«, in: *University of Kansas Science Bulletin* 38, 1409–1438.

Szyperski, Clemens/Dominik Gruntz/Stephan Murer (2002): *Component Software. Beyond Object-Oriented Programming,* London: Addison Wesley.

Thomason, Richmond (Hg.) (1974): *Formal Philosophy. Selected Papers of Richard Montague,* New Haven: Yale University Press.

Turney, Peter D./Patrick Pantel (2010): »From Frequency to Meaning: Vector Space Models of Semantics«, in: *Journal of Artificial Intelligence Research* 37/1, 141–188, DOI: https://doi.org/10.1613/jair.2934 (Zugriff vom 21.02.2018).

Vendler, Zeno (1967): *Linguistics in Philosophy,* Ithaca: Cornell University Press.

Widdows, Dominic/Beate Dorow (2002): »A Graph Model for Unsupervised Lexical Acquisition«, in: *Proceedings of the 19th International Conference on Computational*

Linguistics (COLING '02), Taipei, Taiwan, Aug. 24–Sept. 1, 2002, Stroudsburg, 1–7, DOI: https://doi.org/10.3115/1072228.1072342 (Zugriff vom 21.02.2018).

Wittgenstein, Ludwig (1953): *Philosophische Untersuchungen,* Frankfurt am Main: Suhrkamp.

Wunderli, Peter (Hg.) (2014): *Ferdinand de Saussure: Cours de linguistique générale,* Tübingen: Narr.

Zadeh, Lotfi (1965): »Fuzzy Sets«, in: *Information and Control* 8/3, 338–353, DOI: https://doi.org/10.1016/S0019-9958(65)90241-X (Zugriff vom 21.02.2018).

Zlatev, Jordan (2003): »Polysemy or generality? Mu.«, in: Hubert Cuyckens/René Dirven/John R. Taylor (Hg.), *Cognitive Approaches to Lexical Semantics* (Cognitive Linguistics Research, 23), Berlin: Mouton de Gruyter, 447–494, DOI: https://doi.org/10.1515/9783110219074.447, URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.11.2085&rep=rep1&type=pdf (Zugriff vom 21.02.2018).

Gegenstand der Arbeit ist eine computerlinguistische Modellierung der Bedeutungskonstitution in sprachlichen Einheiten. Ausgehend vom Phänomen der Variabilität sprachlicher Bedeutung wird die Bedeutungskonstitution als ein dynamischer Prozess beschrieben, bei dem sich die Bedeutung sprachlicher Einheiten erst innerhalb lokaler Kontexte konkretisiert. Diese Konzeption eines dynamischen Bedeutungsbegriffs stützt sich auf eine der zentralen Annahmen der Kognitiven Semantik, der zufolge Bedeutungen nicht unabhängig vom Kontext existieren.
Methodischer Leitgedanke ist eine empirisch-experimentelle Herangehensweise an sprachwissenschaftliche Problemstellungen. Die bei der empirischen Überprüfung des Modells zu beachtenden Anforderungen an wissenschaftliche Experimente – Kontrolle, Wiederholbarkeit und Variation – werden durch die softwaretechnologische Umsetzung mittels des linguistischen Komponentensystems Tesla (Text Engineering Software Laboratory) berücksichtigt.
Die Modellierung erfolgt vor dem Hintergrund der Distributional Hypothesis nach Zellig Harris über die algorithmische Erfassung sprachlicher Verwendungsmuster in großen Textkorpora. Auf Basis einer Repräsentation des Bedeutungspotentials durch Vektoren wird die Bedeutungskonstitution als informationsverarbeitender Prozess modelliert, im Zuge dessen eine lokale Anpassung der Vektoren im Sinne einer kontextuellen Aktivierung im Vektorraum erfolgt. In der hier vorgeschlagenen Modellierung sind die Verwendungsmuster damit der entscheidende Informationsträger und -lieferant – mit anderen Worten: ohne Muster keine Bedeutung.

Claes Neuefeind ist wissenschaftlicher Mitarbeiter am Institut für Digital Humanities der Universität zu Köln, wo er 2017 mit der vorliegenden Dissertation promoviert wurde. Sein Interesse gilt der Anwendung computerlinguistischer Methoden in den Geisteswissenschaften, mit Schwerpunkten in den Bereichen Maschinelles Lernen, Künstliche Intelligenz und kollaboratives Arbeiten.

ISBN 978-3-946198-40-6
9 783946 198406

WWW.HUMANITIES-MAP.NET